



Army Materiel Systems Analysis Activity



TECHNICAL REPORT NO. TR-2015-34

**FUSION ORIENTED C4ISR UTILITY SIMULATION
(FOCUS) VERSION 1.0
VERIFICATION & VALIDATION REPORT**

SEPTEMBER 2015

DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited.

IAW Memorandum, Secretary of Defense, 27 December 2010, Subject: Consideration of Costs in DoD Decision-Making, the cost of the study resulting in this report is \$1M.

**US ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
ABERDEEN PROVING GROUND, MARYLAND 21005-5071**

DESTRUCTION NOTICE

Destroy by any method that will prevent disclosure of contents or reconstruction of the document.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

WARNING

Information and data contained in this document are based on the input available at the time of preparation.

TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) SEPTEMBER 2015		2. REPORT TYPE Technical Report		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE FOCUS Version 1.0 Verification & Validation Report				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) John Burghardt, Javier Gestido, Eric Harclerode, Elizabeth Jones, Ankit Joshi				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Director US Army Materiel Systems Analysis Activity 392 Hopkins Road Aberdeen Proving Ground, MD				8. PERFORMING ORGANIZATION REPORT NUMBER TR-2015-34	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The Fusion Oriented Communications, Computers, Command, Control, Intelligence, Surveillance, and Reconnaissance (C4ISR) Utility Simulation (FOCUS) is an entity-level, event driven, stochastic C4ISR simulation. FOCUS simulates C4ISR processes, including sensor performance, tasking and collection; the exploitation and processing of data from all sources, fusion of this information into tracks, and the communication of current predicted tracks to a visual simulation of the Common Operating Picture in a three-dimensional battle-space. Developed by the US Army Materiel Systems Analysis Activity, FOCUS is designed for the analysis of C4ISR's impact on tactical decision making in the Acquisition, Analysis, Intelligence, Test and Evaluation, and Experimentation Modeling and Simulation domains. The purpose of this report is to document the Verification and Validation (V&V) testing results on the algorithms, equations, and data for FOCUS.					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT UNCLASSIFIED	b. ABSTRACT UNCLASSIFIED	c. THIS PAGE UNCLASSIFIED			19b. TELEPHONE NUMBER (include area code)

THIS PAGE INTENTIONALLY LEFT BLANK.

CONTENTS

	Page
LIST OF FIGURES	vii
ACKNOWLEDGEMENTS	ix
LIST OF ACRONYMS	x
1. EXECUTIVE SUMMARY.....	2
2. INTRODUCTION	3
2.1 Purpose.....	3
2.2 Problem Statement.....	3
2.3 Intended Use.....	3
2.3.1 Comparative Sensor Performance Analyses.....	3
2.3.2 Comparative Sensor Coverage Analysis.....	5
2.3.3 Line of Sight and/or Sensor Performance over a Terrain.....	5
2.3.4 Comparative Analysis using Laser Designation Systems.....	6
3. FOCUS OVERVIEW	8
4. VERIFICATION AND VALIDATION (V&V).....	9
4.1 Goals.....	9
4.2 Approach.....	9
4.3 Process.....	11
4.4 Supporting FOCUS Documents.....	13
4.5 Roles and Responsibilities.....	13
5. CODE AND DESIGN.....	14
5.1 Software Design.....	14
5.2 Code Statistics.....	14
6. ASSUMPTIONS AND LIMITATIONS.....	16
6.1 EO/IR.....	16
6.2 MTI Radars.....	16
6.3 SAR.....	17
6.4 SIGINT.....	17
6.5 Laser Designation.....	17
6.6 Platform Movement and Behaviors.....	18
6.7 Environment.....	18
6.8 Communications.....	18
7. DATA.....	19
7.1 Platform Data.....	19
7.2 Sensor Data.....	19
7.2.1 EO/IR Data.....	19
7.2.2 MTI Radar Data.....	19
7.2.3 SAR Data.....	20
7.2.4 SIGINT Data.....	20
7.3 RF Emitter Data.....	20
8. SIMULATION ARCHITECTURE	22
8.1 Performance.....	22
8.2 Simulation Backbone.....	23
8.2.1 Event Management.....	23
8.2.2 Internal Clock.....	24

CONTENTS

	Page
8.2.3 Main Execution Loop.....	25
8.3 Model Input and Output.....	25
8.4 Graphical User Interface.....	26
9. ENVIRONMENT	28
9.1 Digital Terrain Elevation Data (DTED).....	28
9.2 Light Detection and Ranging (LiDAR) Data.....	29
9.3 Buildings.....	29
9.4 Terrain Features.....	30
9.5 Line of Sight.....	30
9.6 Environmental Parameters.....	31
10. ENTITY MOVEMENT AND WAYPOINTS	33
10.1 Platform Movement.....	33
10.2 Unit Formation Movement.....	33
10.3 Waypoints.....	34
10.3.1 Graph Traversal.....	34
10.3.2 Patterns.....	34
10.3.3 Payload Actions.....	35
10.3.4 Transportation.....	35
11. SENSORS/INTELLIGENCE	37
11.1 Moving Target Indicator (MTI) Radar.....	37
11.1.1 Search.....	37
11.1.2 Target Acquisition.....	38
11.1.3 Timing.....	38
11.1.4 Probability of Detection Sensitivity.....	39
11.1.5 Tactical Parameter Sensitivities.....	39
11.2 Synthetic Aperture Radar (SAR).....	39
11.2.1 Search.....	40
11.2.2 Target Acquisition.....	40
11.2.3 Timing.....	40
11.2.4 Probability of Detection Sensitivity.....	41
11.2.5 Tactical Parameter Sensitivities.....	41
11.3 Electro-Optical/Infrared (EO/IR) Imaging Sensors.....	41
11.3.1 Search.....	41
11.3.2 Target Acquisition.....	44
11.3.3 Timing.....	45
11.3.4 Probability of Detection Sensitivity.....	46
11.3.5 Tactical Parameter Sensitivities.....	46
11.4 SIGINT.....	46
11.4.1 Search.....	47
11.4.2 Target Acquisition.....	47
11.4.3 Direction Finding.....	48
11.4.4 Timing.....	49
11.4.5 Probability of Detection Sensitivity.....	49
11.4.6 Tactical Parameter Sensitivities.....	50

CONTENTS

	Page
11.5	Human Intelligence (HUMINT).. ..50
11.6	Laser Designation (LD).....51
12.	COMMUNICATIONS.....54
12.1	Communications.54
12.2	RF Propagation.54
12.2.1	TIREM.....54
12.2.2	Hata Model.55
12.2.3	FOCUS Hybrid Method.55
13.	MISSIONS AND BEHAVIORS56
13.1	Mission Flow Architecture.....56
13.2	Mission Types.56
13.2.1	Cover AOI.57
13.2.2	Identify Track.57
13.2.3	Move.....58
13.2.4	MTI Coverage.58
13.2.5	SIGINT Coverage.....59
13.2.6	Track Target.59
13.2.7	SAR Coverage.....59
13.2.8	Find Track.60
13.2.9	Collect HUMINT.60
13.2.10	Convoy Support.....61
13.3	Conditions.61
13.4	Events.....62
13.5	Mission Behaviors.....63
14.	ANALYSIS TOOLKIT.....65
14.1	Raw Data Types.65
14.2	Operators.66
14.3	Data Export.67
14.4	Analysis Mapping..68
15.	OTHER SIMULATION TYPES.70
15.1	Sensor Coverage Analysis.70
15.2	Terrain Analysis (Line of Sight and Probability of Acquisition).....70
16.	TEST CASE TABLE.....72
17.	CONFIGURATION MANAGEMENT82
17.1	Scrum Agile Software Development Method Overview.82
17.2	Roles.....82
17.3	Process Flow.83
17.4	Artifacts.....83
17.5	Implementation using Team Foundation Server.84
17.6	V&V as part of the Software Development Process.....85
18.	ARMY AND JOINT C4ISR STUDY SUPPORT87
19.	SUMMARY88
19.1	V&V Synopsis.88
19.2	Application.....88

CONTENTS

	Page
19.3 Conclusions.	89
REFERENCES.....	90

LIST OF FIGURES

Figure No.	Title	Page
Figure 1:	Conceptual Model - Sensor Performance Analysis	4
Figure 2:	Conceptual Model - EO/IR Coverage Analysis	5
Figure 3:	Conceptual Model – WAS/Terrain Analysis.....	6
Figure 4:	Conceptual Model - Laser Designation Analysis	6
Figure 5:	V&V Approach.....	11
Figure 6:	Cyclomatic Complexity	14
Figure 7:	Essential Complexity	15
Figure 8:	Modular Complexity	15
Figure 9:	Software Development Process	83
Figure 10:	Product Backlog in Team Foundation Server.....	84
Figure 11:	Work Item in Team Foundation Server	85
Figure 12:	Processes - Documentation - Reviews.....	86

LIST OF TABLES

Table No.	Title	Page
Table 1:	Model Area V&V Processes.....	12
Table 2:	Test Case Description and Results.....	72
Table 3:	V&V Results and Approach Summary.....	88

ACKNOWLEDGEMENTS

The U.S. Army Materiel Systems Analysis Activity (AMSAA) recognizes the following individuals for their contributions to this report.

The author(s) are:

John Burghardt, Combat Support Analysis Division (CSAD), AMSAA

Javier Gestido, CSAD, AMSAA

Eric Harclerode, CSAD, AMSAA

Elizabeth Jones, Logistics Analysis Division, AMSAA

Ankit Joshi, CSAD, AMSAA

The authors wish to acknowledge the contributions of the following individuals for their assistance in the creation of this report:

Stephen Colegrove, Weapons Support Analysis Division, AMSAA

Jaime Howard, CSAD, AMSAA

Heather McMahon-Puhalla, CSAD, AMSAA

LIST OF ACRONYMS

AMSAA	-Army Materiel Systems Analysis Activity
AOI	-Areas of Interest
AR	-Army Regulation
ARL	-Airborne Reconnaissance Low
C2	-Command and Control
C4ISR	-Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance
CDF	-Cumulative Density Function
CM	-Configuration Management
CMIF	-Center for Multi-source Information Fusion
COMINT	-Communications Intelligence
COP	-Common Operating Picture
CPI	-Coherent Processing Interval
CSAD	-Combat Support Analysis Division
CTF	-Contrast Threshold Function
CTI	-Cooperative Target Identification
DTED	-Digital Terrain Elevation Data
DVO	-Direct View Optics
EABICM	-Echelons above Brigade Combat Team Intelligence Collection Mix
ELINT	-Electronics Intelligence
EMARSS	-Enhanced Medium Altitude Reconnaissance and Surveillance System
EO	-Electro Optical
FCS	-Future Combat Systems
FLIR	-Forward Looking Infrared
FMV	-Full Motion Video
FOCUS	-Fusion Oriented C4ISR Utility Simulation
FOR	-Field of Regard
FOV	-Field of View
FW	-Fixed Wing
GPS	-Global Positioning System
GSD	-Ground Sample Distance
GUI	-Graphical User Interface
HFOV	-Horizontal Field of View
HUMINT	-Human Intelligence
I2	-Image Intensifiers
ID	-Identification
IED	-Improvised Explosive Device

INT	-Intelligence
IR	-Infrared
ISR	-Intelligence, Surveillance, and Reconnaissance
IWARS	-Infantry Warrior Simulation
JIEDDO	-Joint Improvised Explosive Device Defeat Organization
LD	-Laser Designation
LEMV	-Long Endurance Multi-Intelligence Vehicle
LiDAR	-Light Detection and Ranging
LOB	-Line of Bearing
LOE	-Level of Effort
LOS	-Line of Sight
MOE	-Measures of Effectiveness
MOP	-Measures of Performance
MRT/MRC	-Minimum Resolvable Temperature/Minimum Resolvable Contrast
MSDF	-Multi-Sensor Data Fusion
MTI	-Moving Target Indicator
M&S	-Modeling and Simulation
NAI	-Named Area of Interest
NIIRS	-National Imagery Intelligence Rating Scale
NVESD	-Night Vision Electronic Sensor Directorate
NVLaserD	-Night Vision Laser Designation
OE	-Organizational Elements
PED	-Processing Exploitation Dissemination
PInf	-Probability of Detection given Infinite Time
P(d)	-Probability of Detection
QA	-Quality Assurance
RAM	-Random Access Memory
RCP	-Route Clearance Patrol
RCS	-Radar Cross Section
RBCI	-Radio Based Combat Identification
RF	-Radio Frequency
RMS	-Root-Mean-Square
ROI	-Return on Investment
RW	-Rotary Wing
SAR	-Synthetic Aperture Radar
SIGINT	-Signals Intelligence
SME	-Subject Matter Expert

SNR	-Signal-to-Noise Ratio
SQ	-Status Quo
SUAS	-Small Unmanned Aerial System
SUNY-UB	-State University of New York – University at Buffalo
TA	-Target Acquisition
TADM	-Target Acquisition Draw Methodology
TADV	-Target Acquisition Decay through Vegetation
TAI	-Targeted Area of Interest
TCM	-TRADOC Capability Manager
TFS	-Team Foundation Server
TLE	-Target Location Error
TIREM	-Terrain Integrated Rough Earth Model
TLS	-Time Limited Search
TRADOC	-Training and Doctrine Command
TRAC	-TRADOC Analysis Center
TTP	-Tactics, Techniques and Procedures
TTPM	-Targeting Task Performance Metric
TV	-Television
UAS	-Unmanned Aerial System
UID	-Unique Identifier
V&V	-Verification and Validation
VPL	-Vertical Plane Launch
VTOL	-Vertical Takeoff and Landing
VV&A	-Verification, Validation and Accreditation
WAS	-Wide Area Surveillance
XML	-Extensible Markup Language

THIS PAGE INTENTIONALLY LEFT BLANK.

FUSION ORIENTED C4ISR UTILITY SIMULATION (FOCUS) VERSION 1.0 VERIFICATION & VALIDATION REPORT

1. EXECUTIVE SUMMARY

This report documents the Verification and Validation (V&V) completed by the Army Materiel Systems Analysis Activity (AMSAA) in order to assess the credibility and accuracy of the algorithms, equations, and data utilized by the Fusion Oriented Communications, Computers, Command and Control, Intelligence, Surveillance and Reconnaissance (C4ISR) Utility Simulation (FOCUS) Version 1.0. FOCUS has been developed by AMSAA as a tool to analyze C4ISR's impact on tactical decision making in the Acquisition, Analysis, Intelligence, Testing, and Experimentation Communities.

FOCUS is designed to simulate the performance of C4ISR systems at levels below brigade and to permit rapid analysis and interpretation of simulation data. Simulation of events down to platform-level resolution (vehicle, aircraft, dismounted soldier, etc.) as well as behaviors such as movement, collection, acquisition, and communications in robust code modules will be used to predict overall performance of C4ISR systems of systems.

As stated in the Army Regulation (AR) 5-11 Management of Army Modeling & Simulation (M&S) [Reference 1], V&V is required for all Army M&S. Verification is the process of determining that the M&S accurately represents the developer's conceptual description and specification. Validation is the process of determining the extent to which the M&S is an accurate representation of the real world. The documentation prepared in the V&V process will support a class accreditation decision. Accreditation is an official certification by the M&S application sponsor that the M&S has achieved a level of fidelity and credibility that it can be used for a specific application or class of applications.

The FOCUS V&V documents the approach, processes, and results of the various methods used to accomplish the following:

- Ensure model implementations are a credible depiction of the real-world
- Show that the simulation functions according to the specification
- Examine model architecture to ensure its stability and flexibility
- Examine physical algorithms and data structure for correctness and maintainability
- Find and correct problems with the model's implementation
- Establish the assumptions and limitations of the simulation
- Ensure documentation exists and is clear and correct
- Ensure Configuration Management (CM) process is established
- Support Accreditation

Through an extensive series of detailed software inspections, unit tests, integration tests, demonstrations and parametric analyses, FOCUS Version 1.0 has been shown to perform as expected within its analytical realm.

2. INTRODUCTION

2.1 Purpose. The purpose of this report is to present the V&V testing results on the algorithms, equations, and data utilized by FOCUS Version 1.0. The report will contain short descriptions of functionality, code architecture, and methodologies; for additional details on these topics, please reference the User's Manual, Design Specification, and Methodologies Specification documents.

2.2 Problem Statement. The AMSAA Combat Support Analysis Division (CSAD) Intelligence, Surveillance and Reconnaissance (ISR) Branch identified a need for an in-house model which could be used as a test-bed to develop and evaluate algorithms related to C4ISR processes and Multi-Sensor Data Fusion (MSDF). There was a demand for a C4ISR model that can be used to evaluate alternative components/methodologies that are part of a "system of systems" based on their relative contribution to the function of the overall system. The model would also use the developed algorithms to perform analysis of real life systems through the creation of scenarios that would incorporate C4ISR and fusion.

FOCUS was built to meet the need identified above. In order to use FOCUS to support and contribute to studies, V&V is required. The V&V effort presented here describes how well FOCUS accurately represents entities at the platform (vehicle, aircraft, dismounted soldier, etc.) and sensor level with respect to behaviors such as movement, collection, acquisition, and communications in operational vignettes. This effort verifies all algorithms and methodologies incorporated into FOCUS.

2.3 Intended Use. This document is intended to show the testing results of the simulation. It shall be used as a reference in the assessment of the validity of the model. It will ensure that the coding is of sufficient quality, contains sufficient internal documentation, responds correctly to commands provided by the user, carries out the mathematical calculations to the required accuracy, and meets the performance requirements.

FOCUS is intended to simulate the performance of ISR systems at levels below brigade, and to permit rapid analysis and interpretation of simulation data. Each section below will identify an application purpose for FOCUS and describe the interactions between the key modules that make up the representation through a conceptual model. Each general use case may include several specific use case threads. A detailed description of the constraints, limitations, and assumptions for the FOCUS methodologies will be provided in future FOCUS Methodology documentation.

2.3.1 Comparative Sensor Performance Analyses. One of the primary purposes of FOCUS is to perform comparative sensor performance analyses between ISR systems across the following domains:

- Electro-Optical/Infrared (EO/IR)
- Moving Target Indicator (MTI) Radar - Ground and Dismount
- Synthetic Aperture Radar (SAR)

- Signals Intelligence (SIGINT) – Communications Intelligence (COMINT) and Electronics Intelligence (ELINT)

The performance analyses are sensitive to the following operational conditions:

- Terrain
- Environmental conditions
- Threat composition and behaviors
- ISR system Tactics, Techniques, and Procedures (TTPs) (e.g., movement, search pattern, cueing)

This use case encompasses all the FOCUS sensor modules and utilizes the full simulation architecture. The specific use case threads include comparing the ability of different ISR systems to:

- Detect targets in an operational vignette
- Identify targets in an operational vignette
- Track targets in an operational vignette

In Figure 1 there are two components that make up an ISR system, the sensor system and the platform. There are six classes of sensors, EO/IR Imaging Sensors, MTI Radar, SAR, COMINT Receivers and ELINT Receivers. The platform carries the sensor system. In the case of FOCUS there are three types of platforms. These are aerial - fixed wing, aerial - rotary wing, and ground. Platforms have a maximum payload in terms of volume and weight, they have a maximum speed, and in the case of the aerial platforms a maximum altitude.

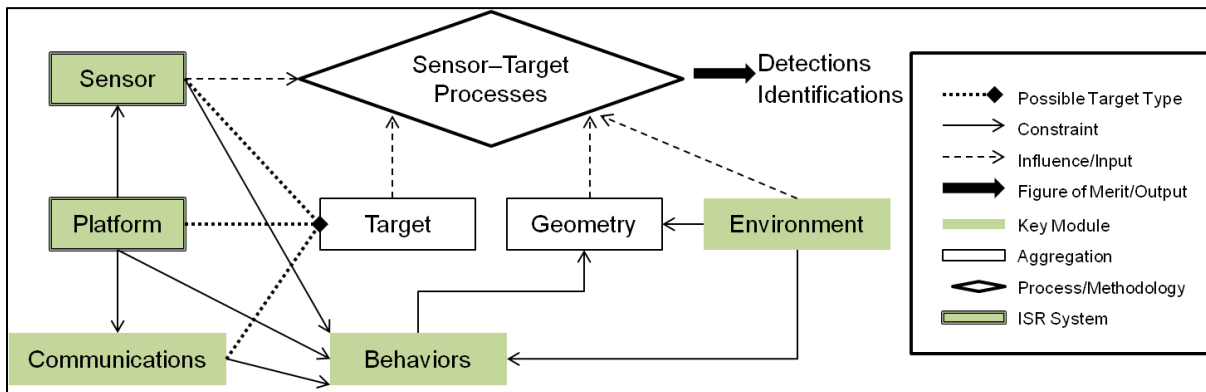


Figure 1: Conceptual Model - Sensor Performance Analysis

The sensors interact with the target, which could be a Radar sensor, platform or communications object depending on the sensor type, and the result of the interaction is detection/no detection or identification/no identification.

The sensor-target acquisition process is influenced by several outside factors. The two shown are geometry (i.e., Line of Sight (LOS) and range), and transient environment (e.g., temperature, rain, snow, turbulence, light levels). The environment is a key module that includes inputs for the sensor-target acquisition processes. The environment also affects the geometry and the behaviors of entities due to terrain. The entity behaviors (e.g., route selection/flight patterns, location/motion of the targets) also affect the geometry between sensor and target.

The figures of merit generated in this purpose include detections and identifications. Two specific purpose threads compare these metrics directly by analyzing the number of detections and identifications of each target or the total number of targets detected or identified. The ability to track a target is measured by analyzing the time between detections to find the percentage of time when a target track may be lost.

2.3.2 Comparative Sensor Coverage Analysis. Another primary application of FOCUS is to perform comparative sensor coverage analysis between EO/IR system(s) given a set of TTPs.

This use case involves executing the following two special use case threads:

- Determine the area covered by an EO/IR system given a set of TTPs
- Determine the number of times an area block is revisited by an EO/IR system given a set of TTPs

Figure 2 depicts the conceptual model for the EO/IR coverage analysis purpose. This purpose is similar to the use case described in Section 2.3.1 except the target is extracted from the process and the figure of merit becomes the footprint of the sensor. This use case is only applicable to EO/IR sensors that have a small field of view compared to the other Wide Area Surveillance (WAS) type sensors.

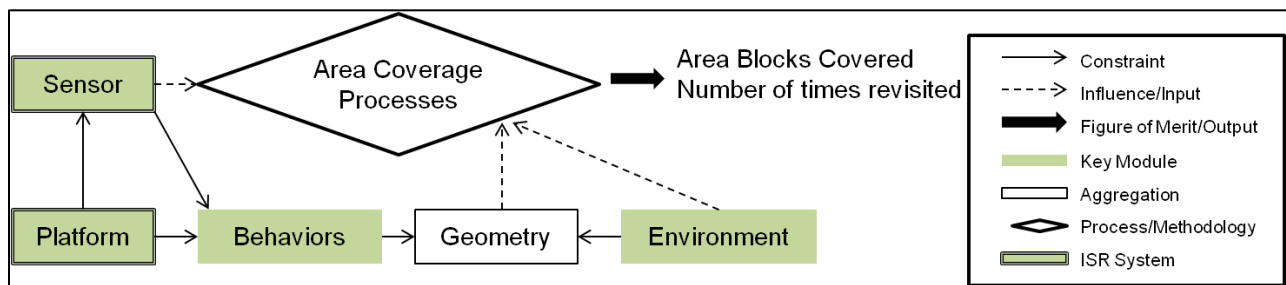


Figure 2: Conceptual Model - EO/IR Coverage Analysis

The ISR system is comprised of the platform and EO/IR sensor. The behaviors of the system and the environment affect the geometry which consequently affects the size of the sensor footprint. The terrain also affects the LOS between the sensor and the footprint area. The sensor field of view characteristics, the geometry, and terrain are processes to give a coverage area. This metric over time will also give the number of times an area is revisited.

2.3.3 Line of Sight and/or Sensor Performance over a Terrain. Another major function is to perform LOS and/or sensor performance analysis over a given terrain and sensor location(s).

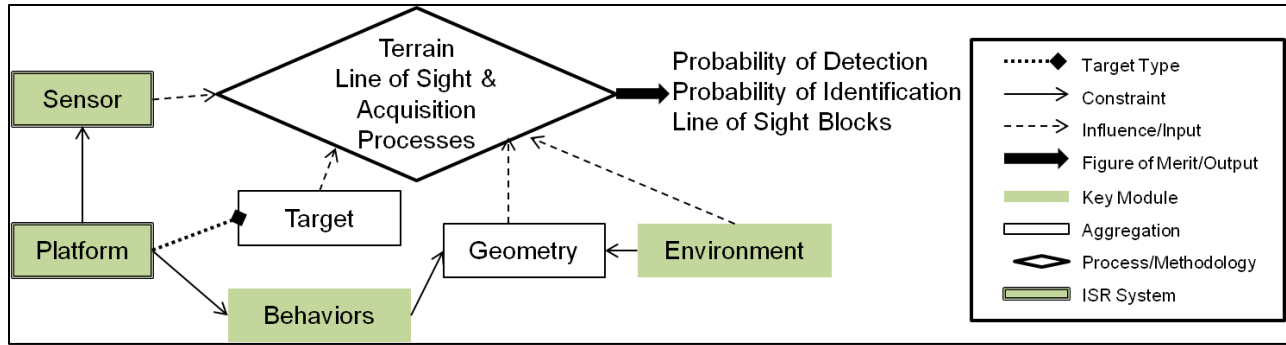


Figure 3: Conceptual Model – WAS/Terrain Analysis

This use case involves executing the following special threads:

- Compare the effectiveness of different ISR system TTPs to detect targets on a complex terrain.
- Compare the effectiveness of different ISR system TTPs to identify targets on a complex terrain.
- Compare the effectiveness of different ISR system TTPs to cover (i.e., have LOS to) a complex terrain.

The WAS/terrain analysis purpose again uses the ISR system (sensor and platform) in conjunction with the environment, platform behaviors, and target object to create a different set of figures of merit. This purpose ignores the sensor behavior and assumes that the entire terrain space is in the footprint of the sensor and thus is more appropriate for WAS sensors. Each terrain point is evaluated for each platform location to determine line of sight and probability of detection and, if applicable, identification of either a person or vehicle target.

The figures of merit include an average probability of LOS/detection/ID for the terrain and platform locations. For this use case, FOCUS also produces a density mapping of the terrain points to depict the average values over all platform locations.

2.3.4 Comparative Analysis using Laser Designation Systems. The final primary use case of FOCUS involves performing a comparative analysis between laser designation systems' ability to achieve target lock-on with a guided munition seeker.

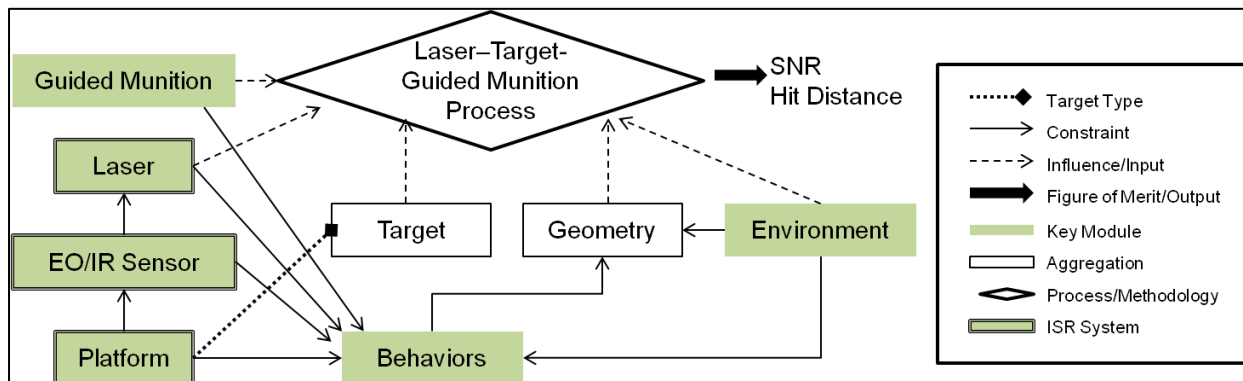


Figure 4: Conceptual Model - Laser Designation Analysis

The laser designation analysis use case introduces two new modules, the laser and guided munition representations. The process uses a target, geometry and environmental parameters. The geometry includes the line between the laser and the target as well as the line from the target to the guided munition. LOS with the terrain is considered for both of these geometrics.

The ISR system also includes the laser paired with an EO/IR sensor. The guided munition has a seeker that searches for the laser spot and then uses a defined trajectory to attack the target. The end metric, hit distance from target ground truth, is affected by the Signal-to-Noise Ratio (SNR) at each time step, which determines whether lock-on has occurred.

3. FOCUS OVERVIEW

FOCUS is an entity-level, event driven, stochastic, ISR-centric simulation. FOCUS simulates ISR processes, including the following: sensor performance, tasking and collection; the exploitation and processing of data from all sources; the fusion of this information into tracks; and the communication of current predicted tracks to a visual simulation of entities and events in a three-dimensional battle-space. FOCUS is developed by AMSAA for use in the analysis of the impact of ISR on tactical decision making in the Acquisition, Analysis, Operations, Testing, and Experimentation M&S Communities.

FOCUS can be used to rapidly assess the performance of ISR systems in small, operational vignettes in complex environments such as urban and mountainous terrains. A typical use case is the comparison of a mix of aerial systems conducting search and tracking missions using Single or Multi-Intelligence (INT) sensors.

The pre-processing vignette builder is a simple-to-use graphical user interface that enables a user to quickly generate a scene, entities, and behaviors using point and click operations. Terrain can be selected from an internal database for low resolution elevation data (30-100 meter intervals) or can be imported from external text files for high resolution elevation data (1 meter intervals). Buildings and other environmental features can be added to the terrain surface.

FOCUS represents entities at the platform (vehicle, aircraft, dismounted Soldier, etc.) and sensor (EO/IR, Radar, SIGINT) level. Behaviors such as movement, collection, acquisition, and communications are defined for each entity by the user when setting up the vignette. Behaviors can either be manually generated by placing waypoints on the terrain or by constructing a flow diagram of built-in, autonomous “missions” along with dynamic conditionals and events. A post-processing analysis toolkit is integrated into FOCUS to filter the output file and extract the desired results. The results can be viewed using the internal FOCUS graphs or exported for further spreadsheet analysis.

Potential applications of the FOCUS model include ISR analysis (sensor mix, unit behavior, fusion effects, urban terrain, sensor cueing, etc.) and TTP comparative analysis. FOCUS has been utilized in several Aerial ISR Mix studies to compare and analyze the systems in urban and mountainous environments by examining the area search and target tracking performance. Sample metrics include, but are not limited to, the number of detections, identifications, and LOS blocks for searching and the target “track” lost intervals over time for tracking.

4. VERIFICATION AND VALIDATION (V&V)

As stated in AR 5-11, Management of Army M&S [Reference 1], V&V is required for all Army M&S. Verification is the process of determining that the M&S accurately represents the developer's conceptual description and specification. Validation is the process of determining the extent to which the M&S is an accurate representation of the real world. The documentation prepared in the V&V process will support a class accreditation decision. Accreditation is an official certification by the M&S application sponsor that the M&S has achieved a level of fidelity and credibility that it can be used for a specific application or class of applications.

In addition to satisfying policy requirements, VV&A can achieve the following:

- Increase confidence in program M&S
- Reduce risk of incorrect decisions
- Increase leverage of existing M&S for future applications
- Achieve better understanding of assumptions and limitations leading to better analysis

4.1 Goals. The goals of the FOCUS V&V are to:

- Ensure model implementations are a credible depiction of the real-world
- Show that the simulation functions according to the specification
- Examine model architecture to ensure its stability and flexibility
- Examine physical algorithms and data structure for correctness and maintainability
- Find and correct problems with the model's implementation
- Establish the assumptions and limitations of the simulation
- Ensure documentation exists and is clear and correct
- Ensure CM process is established
- Support Accreditation

4.2 Approach. V&V of the simulation in the form of test cases was performed using a variety of techniques. Test case descriptions and results can be found in Table 2. The following techniques, defined in [Reference 2] and also depicted in Figure 5, were used to conduct the FOCUS V&V effort:

- Validation Methods
 - *Requirements Specification.* Defines the functional and performance specification of the simulation and sub-models for validation by subject matter experts (SME)
 - *Data/Methodology SME Approval.*
 - Data
 - Verify consistency between data provided and model usage in terms of assumptions, caveats, and conditions
 - Validate the data as representative of some empirical standard to attain consistency and reasonableness
 - Methodology
 - Methodology descriptions for each sub-model can be approved and validated by SME
 - Validate methodology meets a textbook/physics solution or is an analytical community standard

- *Stand-alone Model Benchmarking*. Compares the results of FOCUS methodology to validated stand-alone models that produce the same metrics.
- *Sensitivity Analysis*. Identifies how the sub-models react to varying inputs. Sensitivities ensure the reasonableness and consistency of the model output using validated criteria from one or several of the following methods:
 - Historical real-world or test data
 - Independent review, either by designated committees or by SMEs, to determine if model results are “reasonable”
 - Peer review groups within the study process
- *Demonstration in Study*. Produces fully integrated results from a realistically defined vignette that can be validated by SME and compared to real-world data trends.
- Verification Methods
 - *Unit Testing*. A unit test is a controlled execution of a subset of code that will return a quantifiable result. This result can be compared to an expected result that has been verified by a stand-alone model or by SME judgment. Unit testing is completed in FOCUS by running test code built in the Microsoft Visual Studio Interactive Development Environment. The test code sets up a method call using predefined inputs, executes the method, and compares the result to a hard coded expected value(s).
 - *Integration Testing*. An integration test is a test that does not always have a simple quantifiable result and is the compilation of many methods or models. Integration testing in FOCUS is completed by writing several lines of data to output files during a complex series of method calls or during the running of a live vignette. The data output can then be reviewed for accuracy by an SME based on the methodology specification. Integration tests output lines can be embedded in the code and be enabled/disabled during software execution in the Testing configuration.
 - *Structured Walk-through (Inspection)*. A structured walk-through is a technique that enables all personnel involved to understand expected model outputs by stepping through executing code in order to understand the flow of execution and data. It provides the opportunity for the designer, coder, and reviewer to make a detailed review of the coded algorithm to ensure that the algorithm functions as intended by the modeler and that the necessary dynamic data interactions take place properly.
 - *Stand-alone Model Benchmarking*. Verifies that the results of FOCUS methodologies match the results produced by validated stand-alone models that produce the same metrics.
 - *Sensitivity Analysis*. A method of varying the input in order to analyze the output from a numerical, statistical, and behavioral perspective to determine if the first, and higher, order effects of the algorithm had surfaced as intended by the modeler.
 - *Demonstration in Study*. A demonstration can visually show the result of a methodology or model interaction/process. This technique enables a tester to perform a test using the actual software implementation and verify the actions of a model process.

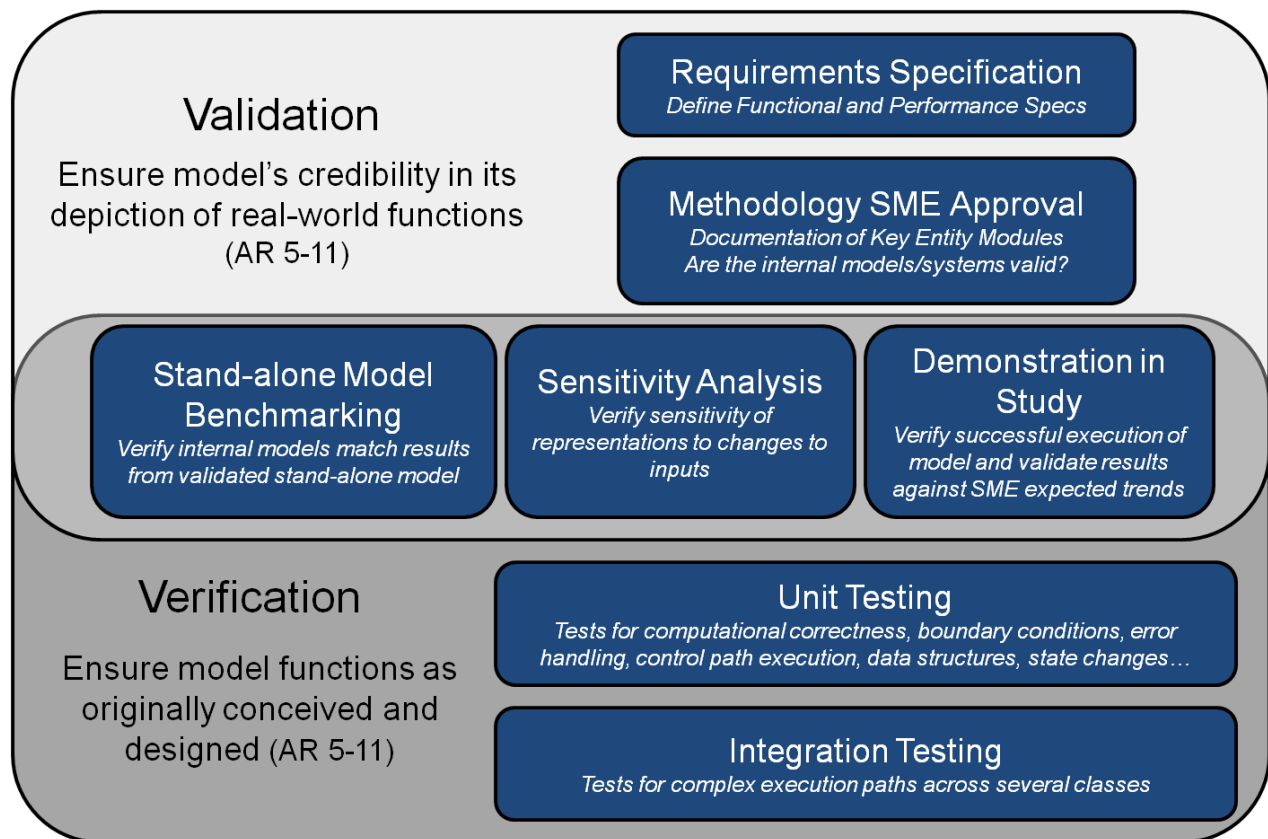


Figure 5: V&V Approach

4.3 Process. V&V is a continuous part of any software development cycle and thus the statements in this report are only applicable for the specific baseline version stated. The FOCUS V&V effort for version 1.0 covers the following critical simulation areas:

- Data Inputs
- Simulation Architecture
 - Performance/Efficiency Checks
 - Internal Clock
 - Event Scheduling
 - Model Input/Output Integrity
 - User Interface Verification
- Environment
- Entity Movement
 - Platforms
 - Waypoints
 - Units
- Sensors/Intelligence
 - EO/IR
 - Laser Designation
 - MTI Radar
 - SAR
 - SIGINT
 - Human Intelligence (HUMINT)

- Communications/Radio Frequency (RF) Propagation
- Missions
 - Mission Flow Sequence
 - Missions Types
 - Mission Behaviors
 - Events
 - Conditions
- Analysis Toolkit
- Mini-simulations

Table 1 illustrates how each key simulation area was assessed as part of the V&V. Note that Multi-Sensor Data Fusion is not included in FOCUS 1.0.

Table 1: Model Area V&V Processes

Model Areas for V&V	General Results	SME Validation	Stand-Alone Benchmark	Sensitivity Analysis	Demo Through Study	Unit Testing	Integration Testing
Simulation Architecture	Pass	n/a	n/a	n/a	✓	✓	✓
Environment	Pass	n/a	n/a	n/a	✓	✓	✓
Entity Movement	Pass	n/a	n/a	n/a	✓	✓	✓
Sensors							
<i>EO/IR</i>	Pass	✓	✓	✓	✓	✓	✓
<i>Laser Designation</i>	Pass	✓	✓	✓	✓	✓	✓
<i>MTI Radar</i>	Pass	✓	✓	✓	✓	✓	✓
<i>SAR</i>	Pass	✓	✓	✓	✓	✓	✓
<i>SIGINT</i>	Pass	✓	✓	✓	✓	✓	✓
<i>HUMINT</i>	Pass					✓	✓
Multi-Sensor Data Fusion	Future						
Communications/ RF Propagation	Pass	✓	✓		✓	✓	✓
Missions	Pass	n/a	n/a	n/a	✓	✓	✓
Analysis Toolkit	Pass	n/a	n/a	n/a	✓	✓	✓
Mini Simulations	Pass	n/a	n/a	n/a	✓	✓	✓

Note: Empty cells indicate the method was not used for the given Model Area

4.4 Supporting FOCUS Documents.

Requirements Specification – Describes the functions of the simulation in terms of what must be implemented without specifying how the functions are implemented.

Design and Architecture Specification – Describes how the functions of the simulation are implemented. This document gives detailed descriptions on the architecture of the code and provides information on the class structure and execution flow.

Methodology Description– Describes the details of all algorithms/methodologies incorporated into FOCUS. This document gives detailed constraints, limitations, and assumptions for use of the simulation. Each methodology in the specification is validated by a SME in that functional area.

User's Manual – Describes how to use the software to setup vignettes within the simulation, execute a study, and extract metrics using the analysis toolkit.

4.5 Roles and Responsibilities. The V&V of FOCUS was performed and sponsored by AMSAA. AMSAA is responsible for the V&V of the critical physical models incorporated within FOCUS. AMSAA will provide the V&V report. Accreditation reports will be generated as studies require.

5. CODE AND DESIGN.

5.1 Software Design. FOCUS is written in C# using the .NET Framework for Windows applications. FOCUS uses an object-oriented modular design approach that allows encapsulation of methods and properties into classes and high levels of code reuse. The object-oriented design allows quick and efficient methodology additions and changes. There are currently 275,000+ lines of code and over 1200 object classes.

5.2 Code Statistics. Statistics on the complexity of the code were produced using the McCabe software. The following metrics were produced for each method in the FOCUS code:

- Cyclomatic Complexity ($v(g)$) – represents the number of paths through the code, ideally less than 21
- Essential Complexity ($ev(g)$) – represents the number of “unstructured” decision logic occurrences (e.g., method return within an “if” statement), ideally less than 5
- Module Design Complexity ($iv(g)$) – represents the number of occurrences of decision logic making subroutine calls, ideally less than 8

All complexity tests show an acceptably low percentage of methods that are overly complex as indicated in Figures Figure 6, Figure 7, and Figure 8. The percentages were calculated by dividing the number of methods over the ideal values by the total number of methods in FOCUS.

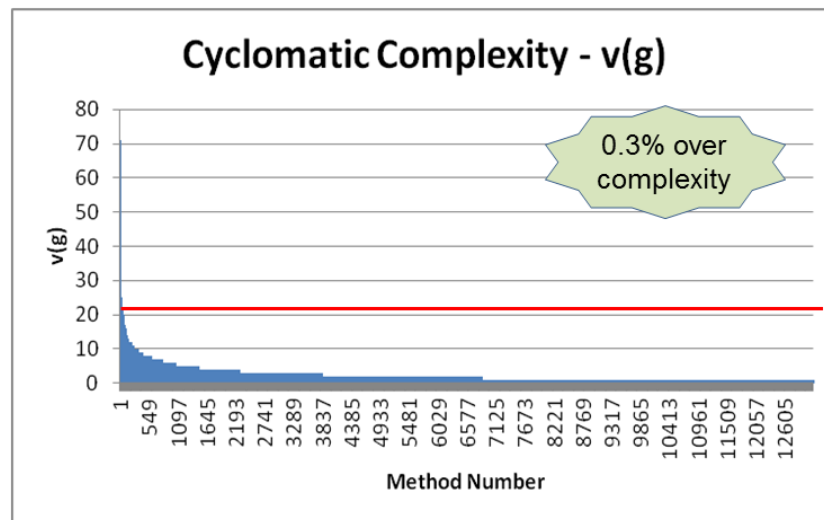


Figure 6: Cyclomatic Complexity

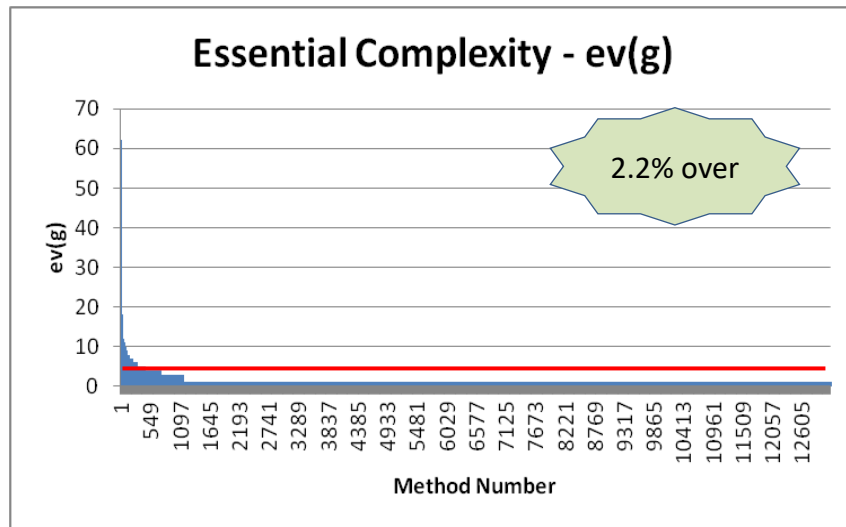


Figure 7: Essential Complexity

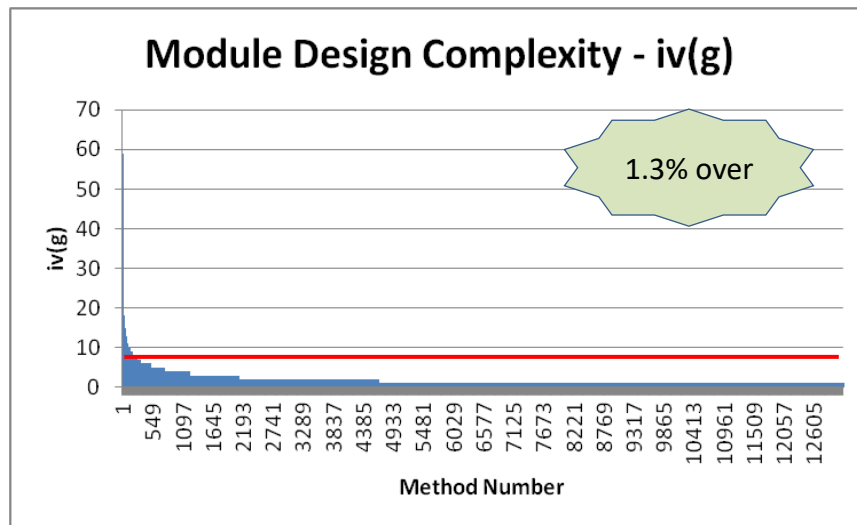


Figure 8: Modular Complexity

6. ASSUMPTIONS AND LIMITATIONS.

This section outlines the high level assumptions and limitations of the primary FOCUS modules.

1. *Assumption* - An assumption is a statement related to the model that is taken as true in the absence of facts, often to accommodate a limitation.
2. *Limitation* - An inability of the study team to fully meet the model objectives or fully investigate the model issues.

6.1 EO/IR.

Assumptions

- Acquire-Targeting Task Performance Metric (TTPM) is a valid representation for determining probability of detection/identification
- The default Field of View (FOV) used is the widest that has a probability of detection (P(d)) for a vehicle target of at least 0.7 unless otherwise set by the user
- The time limited search methodology is a valid representation of timing considerations for search and target acquisition processes
- Each sensor has a dedicated human operator/analyst
- FOV is defined by a horizontal and vertical angle
- The FOV is sampled (checked for possible targets) at a user defined rate while maintaining the rules of the time limited search methodology
- Sensor stays centered on target when tracking

Limitations

- Sensors must have a discrete set of FOV modes
- Discrete set of light levels
- Only model Detection and Identification (ID)
- Extinction Coefficient pulled from a range lookup table for each waveband and environmental condition, values not in table are interpolated

6.2 MTI Radars.

Assumptions

- Golden Point Method is valid representation for MTI detection
- Uses target radial velocity to compare to minimum detected velocity
- Simple search field of view uses current pointing angle with horizontal beam width and full vertical angles to determine if target is in view
- Simple search method moves beam width based on defined scan rate
- Complex search field of view uses full horizontal and vertical angles to determine if target is in view
- Targets are randomly chosen to have a dwell time in field of view during complex search

Limitations

- Valid for Swerling one and two targets only
- A target is represented by a single radar cross section (RCS) value

6.3 SAR.

Assumptions

- National Imagery Intelligence Rating Scale (NIIRS) to P(d) methodology is valid representation
- Image exploitation times follow normal distribution
- Targets are identified only after completion of a strip
- Collection for each block is a fixed time value (user defined)
- Wide mode FOV used for detections, Narrow mode FOV used for identifications
- Each strip of imagery has a dedicated analyst
- Multiple detections in a strip are separated by one second intervals

Limitations

- Only two resolutions (i.e., FOVs) modeled (wide and narrow)
- Search conducted using strip search only with spot mode identification
- Square FOV dimensions for each resolution mode

6.4 SIGINT.

Assumptions

- One-way range equation with sensitivity is valid representation for probability of detection
- ELINT receivers can be modeled with a fixed horizontal FOV angle or use a gimbal speed (scan rate) to determine horizontal area covered
- ELINT receivers are not modeled with a vertical FOV restriction
- COMINT receivers are Omni-directional

Limitations

- Direction finding capability uses a weighted lines of bearing method only
- Detection of signals only, does not measure ability to interpret signals
- All SIGINT receivers work together to perform direction finding
- Lines of bearing are combined when 50 are collected or two minutes have passed

6.5 Laser Designation.

Assumptions

- NVLaserD is a valid representation to find the signal-to-noise ratio (SNR) between laser and guided munition
- Guided munition will go to last known laser spot point
- Laser can reflect off any object to guided munition
- If target is in paired EO/IR sensors footprint, the laser spot is on the target (if there is line of sight)
- If laser is not capable of achieving a high enough SNR at the safe range, the laser platform can be closer than the accepted safe range
- Lock-on achieved with SNR above a predetermined threshold

- During descent phase of missile trajectory, missile will follow spot until it hits the ground as long as spot is in seeker FOV

Limitations

- Only single missile type modeled
- Lock on after launch low and high trajectories modeled
- Velocity is determined from fixed equation based on time
- Trajectories are determined based on fixed equations for each phase: launch, search, track, and descent; approximate real data from set ranges

6.6 Platform Movement and Behaviors.

Assumptions

- Platforms do not block line of sight

Limitations

- Velocity set at each waypoint, no acceleration modeled
- Roll, pitch, and yaw not modeled
- Physical dimensions modeled as rectangular block
- Discrete set of automatic behaviors

6.7 Environment.

Assumptions

- Environmental parameters stay constant for the entire vignette

Limitations

- Terrain size limited to approximately 2000 x 2000 points at any given resolution
- Terrain points equally spaced
- No wind effects

6.8 Communications.

Assumptions

- Aerial platforms are assumed to have connection to ground station
- Communications are perfect unless given a message completion rate

Limitations

- Processing Exploitation Dissemination (PED) tasks are not modeled
- Radios have single channel

7. DATA

Data validation is necessary to verify the consistency between data provided and model usage in terms of assumptions, caveats, and conditions as well as to verify correctness. These data are most notably characterized within the simulation as defining attributes of platforms, sensors, and RF emitters.

These data are modified and replaced as appropriate given study demands and updates. Any FOCUS user should ensure that the platforms, sensors, and emitters being studied are represented using the latest available data.

7.1 Platform Data. Platforms are battle-space entities representing a particular individual piece of military equipment. To accurately represent this entity, the model must maintain real-world physical attributes (signatures) for each platform in addition to performing realistic movements in accordance with different terrain. The following characteristics are the essential platform parameters necessary for a valid representation:

- Physical dimensions (Length, Width, and Height) [meters]
- Modulation Contrast [°C]
- Thermal Contrast [°C]
- Radar Cross Section [m²]
- Entity Type (Personnel, Vehicle, Aerial)
- Operational Altitude (Aerial only) [meters]
- Operational Velocity [meters/second]
- Turning Radius [degrees]
- Climb Rate (Aerial only) [degrees]

7.2 Sensor Data. There are several different types of sensors within FOCUS. For an accurate representation, each different type of sensor has its own set of attributes and methods of scanning, collection, detection, and acquisition. The sensor types represented include imaging sensors, radars, and SIGINT.

7.2.1 EO/IR Data. The following characteristics are the essential sensor parameters necessary for a valid representation:

- FOV Modes
 - Acquire-TTPM Shortcut Tables
 - Horizontal and Vertical Angles [degrees]
 - Magnification
- Gain Parameter (IR only)

7.2.2 MTI Radar Data. The following characteristics are the essential sensor parameters necessary for a valid representation:

- Minimum Sensor Range [meters]
- Maximum Sensor Range [meters]
- Minimum Elevation Angle [degrees]
- Maximum Elevation Angle [degrees]
- Minimum Azimuth Angle [degrees]

- Maximum Azimuth Angle [degrees]
- Minimum Detectable Velocity [meters/second]
- M of N – m number of detection instances required out of last n attempts needed for a valid detection
- Pulse Repetition Frequency [Hertz] – the most probable or average value
- Number of Pulses Integrated (pulsed radar only)
- Probability of Detection Method (Golden Point Method, ref = Radar reference value)
 - Range ref [meters]
 - False Alarm Probability ref
 - Probability of Detection ref
 - RCS ref
- Scan Pattern Method (Simple or Complex)
 - Simple
 - Scan Rate [degrees/sec] – rate at which radar scans a sector
 - Beam Width [degrees] – width of the scan beam
 - Complex
 - Scan time [seconds] – time between dwells on target
 - Dwell time [seconds] – time that target is within radar beam

7.2.3 SAR Data. The following characteristics are the essential sensor parameters necessary for a valid representation:

- Block collection time [seconds]
- Gimbal Speed [degrees/second]
- Exploitation Delay Time Distribution
- Maximum Detected Velocity [meters/second]
- Wide FOV
 - NIIRS Rating
 - Block Size [meters]
- Narrow FOV
 - NIIRS Rating
 - Block Size [meters]

7.2.4 SIGINT Data. The following characteristics are the essential sensor parameters necessary for a valid representation:

- Angular Target Location Error [degrees]
- System Sensitivity
- False Alarm Probability
- Minimum and Maximum Detectable Frequencies [MHz]
- Lines of Bearing/Second
- Scan Pattern
 - Horizontal Field of View Angle [degrees]
 - Or
 - Scan Rate [degrees/second]

7.3 RF Emitter Data. Radars and Radios are the RF emitter types in FOCUS. The following characteristics are the essential emitter parameters necessary for a valid representation:

- Power Transmitted [Watts]
- Frequency [MHz]
- Gain [dB]
- Loss [dB]

8. SIMULATION ARCHITECTURE

The simulation architecture is the fundamental backbone of FOCUS. It controls the sequence and timing of events and the integration of all the capabilities of the simulation. Entity behaviors are established and managed to provide an accurate representation of real-world scenarios and objects. An accurate representation is defined by the capability to preserve data integrity, to perform efficient invoking and execution of events, and to provide proper data and time management between these events. Areas within the simulation architecture of the FOCUS model that have been verified and validated are outlined in the sections below.

8.1 Performance. Throughout the FOCUS development effort, run-time checks were conducted to ensure the efficiency of the model and to correct any areas that may have caused a failure in that efficiency. The following tests were executed to test the performance and efficiency of the simulation under normal and stressed conditions:

Test: Runtime Speed Efficiency

Purpose: Examine the relationship between increased scenario complexity and expected run-time.

Objective: Gain an understanding of how the model performs under increasing complexity. Complexity is defined as the use and amount of parameters that would be expected to increase run-time performance such as the number of moving entities and the number of sensors performing search and collection. The goal is to determine how (linear, non-linear) run-time is related to increasing complexity. Parameters were increased over a range of reasonable values and the effect on run-time was recorded.

Type: Integration

Method: Demonstration

Description: A simple scenario was constructed involving platforms with sensors performing a mission to search for moving targets. The number of platforms/sensors and the number of moving targets were increased to see the impact on run time. The vignettes were run for five minutes and ten replications each with the graphics turned off.

Results: For a vignette with the greatest complexity (50 targets and 10 platforms/sensors), FOCUS ran at a satisfactory rate (11.16 x real time) and overall, the runtime increased at a slower than linear rate with the increased entities.

Recommendations/Updates: Overall, the speed at which FOCUS can run is satisfactory but there are opportunities for improvement in several modules. The development team will continue to refactor the code in order to improve run time by using object oriented principles and by inserting processes that can allow the user to change the resolution at which certain algorithms are executed.

Test: Memory Stress (Terrain, Areas of Interest (AOI), Buildings, Platform/Sensor, Missions)

Purpose: Examine the processes/entities that are the most resource intensive and whether they can be adjusted to use less resources.

Objective: Gain an understanding of the limits of scenario complexity. Entities that can have a large number of instances in a scenario should be investigated in terms of memory usage such as the number of terrain points, AOI, buildings, platform/sensors, and missions.

Type: Unit/Integration

Method: Test/Demonstration

Description: Create a series of scenarios that increase the number of entities to see memory overload handling (if it occurs) and the general responsiveness of the software when dealing with the increased memory usage.

Results: Available memory to store the entity data may fluctuate depending on machine specifications and various background operations being executed. For a typical computer with 4GB of RAM, it is recommended to use a terrain with no more than 4 million terrain points (such as a 2 km x 2 km terrain with 1 meter resolution). The number of AOI is not an issue due to the reuse of reference points created for the entire terrain. Building, platform, sensor and mission entities use a small amount of memory each and the model can easily handle a significant amount of entities. As with any graphics model, as the number of entities increases the render time also increases. For building entities, the model allows the user to merge the entities into the elevation data which can dramatically increase render time.

Recommendations/Updates: The terrain size issue is a significant limitation of the model that will be addressed for the next version of the simulation. The current terrain data structure uses a contiguous array that at a certain point cannot fit into memory. An array was used for the speed advantage it holds when accessing points. A new data structure must be used that divides the terrain into smaller chunks that can be separated in memory without a dramatic decrease in speed.

8.2 Simulation Backbone. The simulation backbone is the underlying simulation loop and data structure that controls the sequence of the events/actions and the run time of the software. This backbone includes the event list structure, the internal clock, and the main execution loop.

8.2.1 Event Management. FOCUS is an event driven simulation model. All events that are called are handled by the *Event Manager*. This is a control class that manages all events that will be executed during the simulation run by holding the events in a sorted linked list. The class controls the execution of the events and keeps track of the timing performance.

Test: Event Scheduling

Purpose: Ensure the event manager keeps the scheduled events in the correct order.

Objective: Prove that the event manager posts events in chronological order.

Type: Unit

Method: Test

Description: Post a series of events to the event manager with varying fire times and ensure that they are sorted into the correct sequence.

Results: The events were posted in the correct sequence in all conditions. Events past the simulation end time were not posted as expected.

Recommendations/Updates: The event manager was updated during the V&V when it was discovered that the current data structure had memory issues because the linked list of events was converted to an array during certain processes. The processes that used the array were converted to use the linked list and the memory issue was solved.

8.2.2 Internal Clock. FOCUS executes scenarios with various options set by the user to determine parameters for the simulation run time environment. As part of the V&V process of the FOCUS model, the real time and multiple of real time run mode capabilities were verified that they work correctly and as expected. This primarily involved the verification of the FOCUS internal clock implementation. The internal clock implementation used in the FOCUS model can be found in the *Simulation Timer* class. This is an entity class that represents the internal simulation clock that can be started, paused, and resumed during the replication runs. This class utilizes a timing class which holds the time data structure and controls any operations that can manipulate the time. The key facets that were verified are the *Start*, *Pause*, *Resume*, and the *Time* methods of the *Simulation Timer* class.

Test: Real Time Clock Execution

Purpose: Ensure that the internal clock can keep the execution of events at the correct pace when run in a multiple of real time.

Objective: Prove that a simulation executes events at the user controlled multiple of real time.

Type: Unit

Method: Test

Description: Post a series of events at varying times, execute the model using varying multiples of real time, and examine the real time execution times.

Results: The internal clock kept the execution times consistent with the real time multiplier when possible. When the complexity of the scenario is too high, the simulation runs at the maximum possible speed which may be lower than the multiplier. When the simulation is paused and resumed, the timer adjusts for the paused times.

8.2.3 Main Execution Loop. The main loop of the simulation executes the events and actions, updates the replication and/or simulation variables, and ensures the correct reset of variables between runs.

Test: Simulation Loop

Purpose: Test if the simulation successfully completes multiple replications by resetting the scenario and possibly using the simulation variables.

Objective: Show that the simulation loop can handle multiple replications and simulation variables with the correct scenario start data loaded for each run.

Type: Integration

Method: Demonstration

Description: Run a series of the scenarios with different replication numbers, times, and simulation variables.

Results: Through the use of FOCUS in several studies and utilizing historical results, this test has been verified to pass.

Test: Action Event Flow

Purpose: Examine whether the action event types run as intended including parent updates and children force ending.

Objective: Actions are event types that can have dependents (child actions) that determine when actions end or new actions begin. This test will show that the actions work together to make a series of events that are dependent upon the completion of each other.

Type: Integration

Method: Test/Demonstration

Description: Run a series of scenarios that use actions to complete behaviors. Output the times that actions are created and finished and how the parent/child actions interact.

Results: The results confirm the sequence of events that actions complete when executed. It involves child actions within child actions that complete in a sequence with parent action updates and the completion of a root action that ends the remaining child actions.

8.3 Model Input and Output. File input and output is an important feature of the FOCUS model that allows the user to save data for use at a later time. FOCUS generated files include the scenario, terrain, entity libraries, simulation output and analysis toolkit. As part of the V&V process of the FOCUS model, all input/output files were verified for data integrity.

Test: File I/O (Scenario, Terrain, Output, Library, Analysis)

Purpose: Examine the saving and loading of FOCUS generated files (scenario, terrain, library, output, analysis) to/from an external file.

Objective: Demonstrate that the files are saved and loaded successfully with all variables stable.

Type: Integration

Method: Test/Demonstration

Description: All files were tested by setting variables and saving the file, then loading the file and checking the value of the variables.

- Scenario –The serialization and de-serialization methods from the *Scenario* class were examined. Also, the registry serialization and de-serialization methods of the *Object Registry* class were examined. The approach was to check the consistency of data between the saved scenario, the scenario file, and the loaded scenario.
- Terrain –The methods *Save* and *Load* methods from the *Environment* class were examined. The approach was to check the consistency of data between the saved and loaded terrain.
- Library – The template classes are saved directly using Extensible Markup Language (XML) serialization and each template type was examined to ensure the correct data was serialized.
- Output – The output file consists of a series of record types for each type of analysis data. Each record type was examined for both binary and XML saving/loading. Reading of the records for use in the analysis toolkit is part of the analysis V&V below.

Results: All external files save and load correctly

Recommendations/Updates: The library external files do not have filenames that give a description of what the template is in the file. The next version will name the file after the template name. These templates will also not be opened unless they are being used in the scenario to save memory. Currently, all templates are opened upon software launch.

8.4 Graphical User Interface. The FOCUS graphical user interface (GUI) was designed to enable the user to quickly create and edit scenarios for time sensitive studies. Many of the operations are point and click, drag and drop type interfaces where there is limited data that needs to be typed. Then all components of the GUI were tested for functionality, error handling, and data input.

Test: GUI Functionality

Purpose: Check GUI functionality consistent with the user's manual and does it provide proper error handling.

Objective: Demonstrate that the GUI performs the functions of the software and determine whether it does not allow erroneous input while providing instructions for correcting errors.

Type: Integration

Method: Test/Demonstration

Description: Each GUI component was examined and executed for functional testing. Components with data entry were given erroneous input to test for error handling.

Results: All GUI components are functional. Some inputs did not handle erroneous input and were corrected. Other errors were handled and alert the user but do not give a description of what went wrong.

Recommendations/Updates: Improve the error handling alerts given to the user so the errors can be identified and fixed.

9. ENVIRONMENT

FOCUS must represent real world terrain data and other various environmental conditions and entities in order to accurately simulate the operational vignettes for studies. Environmental parameters, buildings and terrain data, along with its coordinate conversion within the model, were verified and validated in accordance to real world data. The terrain data types in FOCUS include Digital Terrain Elevation Data (DTED) and Light Detection and Ranging (LiDAR) data that are imported from external files. In addition to the physical representation, processes such as line of sight checks were verified.

9.1 Digital Terrain Elevation Data (DTED). The DTED format is available in several resolutions and FOCUS is currently capable of reading Level 1 (~100 meter resolution) and Level 2 (~30 meter resolution). FOCUS has a GUI that allows the user to select a region on a map of the globe and retrieve the DTED files from an organized file system database. The DTED format contains points that take into account the curvature of the earth. In order to be used in the “flat earth” FOCUS terrain format, a conversion algorithm must be used. The tests for DTED include selecting regions from the GUI or manually entering latitude/longitude regions, the reading of the DTED files, and the conversion of the data to a flat equally spaced grid of elevation points.

Test: DTED Import

Purpose: Check that the user defined area/points are correctly utilized to retrieve the appropriate DTED files and if the interface handles the cropping of data based on user selection.

Objective: Verify that the region selected on the GUI and the points entered manually are used to retrieve and crop the correct DTED files and elevation data.

Type: Integration

Method: Test/Demonstration

Description: Each test will use a known location to compare to what is imported by the model. The image produced by the selected regions should match.

Results: The test shows that the software successfully pulls the correct data and produces an image for the GUI.

Test: DTED Coordinate Conversion

Purpose: Ensure the DTED file points are converted to an equally spaced grid that is consistent with the real terrain data.

Objective: Verify that the latitude/longitude DTED data is successfully converted to the equally spaced Cartesian coordinate system used in FOCUS as well as other coordinate systems.

Type: Unit

Method: Test

Description: Exercise the coordinate conversion algorithm on a known set of data to verify the conversion from one system to another. Compare a terrain set with data generated by the geospatial SMEs for consistency.

Results: The coordinate conversion algorithm successfully converts all data points that are within its realm (i.e., all points not close to the earth's poles). The terrain set elevation points contained errors when compared to the geospatial SME data, but it was to be expected because of different smoothing algorithms and was low enough for the purposes that the terrain would be used.

9.2 Light Detection and Ranging (LiDAR) Data. LiDAR data is preprocessed using ArcGIS into a text file that FOCUS is capable of importing. The data in the file is already equally spaced for direct use as a FOCUS terrain. The LiDAR data file is usually too large to completely be used so the user must choose a smaller region. The tests conducted will verify the selection of a smaller region from a GUI and the actual import of the data into the terrain format of FOCUS.

Test: LiDAR Terrain Data Import

Purpose: Test that the LiDAR data is correctly imported into FOCUS and if the user can select a smaller region using the GUI.

Objective: Verify that the LiDAR text file is consistently imported and that the GUI selects the correct subset of data for the user defined terrain space.

Type: Unit and Integration

Method: Test/Demonstration

Description: Create a simple text file in the LiDAR format with known data and import into FOCUS. Compare the terrain elevation data structure to the known data. Using the GUI, select and compare regions that can be compared visually for correctness.

Results: The tests verify the import of the LiDAR files and the demonstration of the GUI verifies the selection of smaller regions of data for use in the scenarios.

9.3 Buildings. The building objects in FOCUS can either be left as individual entities that can be edited or merged into the elevation data. Either way that they are implemented by the user, the building heights will be incorporated into the terrain data for use in line of sight checks.

Test: Buildings in Line of Sight Calculations

Purpose: Test if the building heights are used in line of sight calculations correctly and efficiently.

Objective: Verify that the building heights are merged into the terrain point data and thus are used in line of sight calculations.

Type: Unit

Method: Test/Demonstration

Description: Create buildings with known size and locations and merge the data into the terrain. Compare the new data with the known data.

Results: Through demonstration and file output, it can be shown that the buildings are merged into the terrain at the terrain resolution. The test also verified that the buildings are merged into the correct locations.

Recommendations/Updates: Update the building coordinate system so that they can be used even when the resolution of the terrain is low.

9.4 Terrain Features. Terrain features are three dimensional volumes in a variety of shapes that can be placed on the terrain to represent features such as areas of rain, fog, smoke, vegetation, etc. These features will affect different aspects of the model such as line of sight or the transmission of signals in the sensor modules.

Test: Terrain Feature Hit and Effect

Purpose: Verify that the ray tracing algorithm correctly identifies whether a terrain feature was hit, at what point, and how far into the feature the destination is.

Objective: Verify that the terrain feature implementation as a mesh object can return the required information as described in the purpose.

Type: Unit

Method: Test

Description: Create terrain feature with a known size and location and set a ray through the feature with no start and end points. Verify the hit point and distance through the feature.

Results: The ray tracing algorithm correctly identifies the hit point and the distance through the feature.

Recommendations/Updates: If the user overlaps features that have the same effect type, the algorithm will combine the effects which may not be the intended results. The model should implement some sort of check/alert to notify the user of this consequence.

9.5 Line of Sight. The line of sight calculation is a significant part of all sensor modules that can dramatically change results of a study and must be correct and efficient.

Test: Terrain Line of Sight

Purpose: Check to see if the line of sight algorithm correctly identifies whether there is an obstruction between two points in space.

Objective: Verify the implementation of the line of sight algorithm at different resolutions. The user can change the resolution in meters of the line of sight check to change execution speed.

Type: Unit

Method: Test

Description: Create a terrain with known obstruction points and set two points to check the space between. Change the resolution and points to check different conditions.

Results: The line of sight algorithm correctly identifies whether there are obstructions between two points using the user defined resolution.

9.6 Environmental Parameters. There are several environmental parameters that are used in calculations for the sensor modules. Some of the parameters use the diurnal cycle to change the value over time. Other parameters are calculated based on the range between points in the environment such as the extinction coefficients and the target acquisition decay through vegetation methodology.

Test: Diurnal Cycle

Purpose: Ensure that the parameters that use the diurnal cycle return the correct value for a given time.

Objective: Verify the implementation of the diurnal cycle data value calculation.

Type: Unit

Method: Test

Description: Create a diurnal cycle parameter with known values over time and query the function to get the correct value at a given time.

Results: The diurnal cycle parameters successfully give the correct value whether implemented as piecewise or interpolated.

Test: Extinction Coefficients

Purpose: Verify that the extinction coefficients are calculated correctly for the given range through the environment.

Objective: Verify the calculation of the extinction coefficient either using a static value for a given waveband, the “a-b” method, and a lookup table.

Type: Unit

Method: Test

Description: Exercise the extinction coefficient methods using the different methods to return a known value for a range and waveband combination.

Results: The extinction coefficient calculation for all methods returns the correct value.

Recommendations/Updates: The best method for getting the extinction coefficient is using the range lookup table, but the database is currently hardcoded into FOCUS. Allow the user to change the database file.

Test: Target Acquisition Decay through Vegetation (TADV)

Purpose: Ensure that the TADV methodology returns the correct degradation factor for a given range and posture.

Objective: Verify the TADV methodology correctly accesses the database and calculates the degradation factor given the postures, range, and density.

Type: Unit

Method: Test

Description: Query the database with various inputs and return known values. Calculate the degradation factor and compare to known values.

Results: The TADV methodology successfully returns the correct values from the database and calculates the degradation factor.

10. ENTITY MOVEMENT AND WAYPOINTS

10.1 Platform Movement. Steps taken to mimic realistic platform (ground and air) movement involve calculating distances, direction, turning, and waypoint locations. This modeling primarily involves basic vector addition, subtraction, multiplication, and normalization between the moving platform and the next waypoint destination. However, when calculating platform turning angles, velocity, and velocity direction, more complex equations are used to provide realistic results. These complex movements for both air and ground platforms were verified where results can be found in the following test cases:

Test: Platform Movement (Air & Ground)

Purpose: Verify that the platforms move to the correct location at each time step using the turn radius, climb rate, and velocity.

Objective: Verify the movement of platforms between locations with respect to time, speed, and movement constraints.

Type: Unit

Method: Test/Demonstration

Description: Calculate the movement locations of platforms using various speeds, locations, turning radii, and climb rates. Compare the resulting movement locations against the maximum speed and turning radius to ensure that the movement was within the threshold. Create a sequence of turns to ensure that the platforms can make every possible type of movement in all quadrants.

Results: The movement algorithm successfully moves the platforms with respect to time and movement parameters. The platforms were capable of making each turn type.

10.2 Unit Formation Movement. Units, constructed as Organizational Elements in FOCUS, are abstract categories within the model for platforms and Improvised Explosive Devices (IEDs) to be placed into. Units can represent everything from the entire Order of Battle down to small units (e.g., platoon, terrorist cell). Pre-made organizational element groups (templates) are available for use in scenarios. Groups of platforms under an organizational element are able to move in formation patterns. As part of the V&V process of the FOCUS model, the physical attributes of organizational elements such as formations and group data, as well as their behaviors needed, were verified in accordance to real world data.

Test: Unit Movement using Formations

Purpose: Check that the units move into the correct formation and move along the path between two waypoints.

Objective: Verify the calculation of each formation type and the sequence to form the formation and move to the next waypoint.

Type: Integration

Method: Test/Demonstration

Description: Create the formations and measure the distance between the platforms that make up the formation. Demonstrate the formation and movement along the path in a simple scenario.

Results: The formations are successfully created and the units will move along the path in the formation.

Recommendations/Updates: At each waypoint, the platforms must adjust to the next direction and do not follow a smooth path when the formation stays the same. This should be updated for the next version.

10.3 Waypoints. Waypoints are locations in the terrain environment to which platforms and other moveable objects will move while following some behavior. Waypoints appear in a connected form to show the path of the moving object. As part of the V&V process of the FOCUS model, waypoint data, patterns, and behavior were tested and verified.

10.3.1 Graph Traversal. The waypoints are structured in a graph where there can be split paths and repeating patterns. The platforms/units that traverse the graph must make decisions at waypoints to follow which path. This test would also verify the decision that a platform has reached a waypoint and will now go to the next waypoint.

Test: Waypoint Graph Traversal

Purpose: Verify that the moving objects follow the correct path and move to the next waypoint when they reach the current waypoint.

Objective: Verify the decision to follow a given path and ensure that objects will always be able to reach a waypoint in order to traverse the graph.

Type: Integration

Method: Test/Demonstration

Description: Create a waypoint graph with different decision points and demonstrate that the object can traverse the graph using the known path. Use various speeds and turning radii to test the ability to reach a waypoint in order to move from point to point.

Results: The test verified that objects can successfully traverse a waypoint graph and that the decision paths correct determine the path. The traversal action also correctly identifies when an object has reached a waypoint.

10.3.2 Patterns. Waypoint patterns are predefined sequences of waypoints such as ellipses, figure eights, and zigzags. The test for these patterns verifies the waypoint placement of the different pattern types.

Test: Waypoint Pattern Creation

Purpose: Ensure that the waypoints that make a pattern are correctly placed in order to form the proper sequence.

Objective: Verify that the waypoint patterns are constructed given the user defined parameters and match the pattern sequence they are intended to follow.

Type: Unit

Method: Test/Demonstration

Description: Create waypoint patterns using a variety of input parameters and compare the waypoint locations to see if they match the inputs. Visually inspect the pattern that is formed.

Results: The waypoint patterns are correctly created using the inputs and can be traversed by objects successfully.

10.3.3 Payload Actions. Payload actions are waypoint defined behaviors for payload objects such as sensors and radios. The actions define when a payload is activated and in the case of a sensor, where to point the sensor. A payload action defined at a waypoint is active until it is disabled by another waypoint. The actions can either be per waypoint or per path.

Test: Payload Behaviors set at Waypoints

Purpose: Test whether the payload actions are activated for the correct waypoint and if they stay active until changed by another waypoint.

Objective: Verify the use of the input data when executing an action and the continued use of the action until another action is active at another waypoint.

Type: Integration

Method: Test/Demonstration

Description: Create payload actions with a variety of inputs along a sequence of waypoints to how the payload behaviors are updated as the object reaches each waypoint.

Results: The payload actions are successfully executed using the user defined input parameters and the actions stay active until another action is activated at another waypoint. The actions can be activated by the specific waypoint or the path chosen.

10.3.4 Transportation. Platforms have the ability to load/unload other objects at waypoints. At a specified waypoint, the platform can drop off an object, wait for an object and pick up an object.

Test: Transportation Loading/Unloading at Waypoints

Purpose: Examine whether the load/unload operations are successfully executed as a platform traverses a waypoint path.

Objective: Verify that a platform traversing a waypoint path can unload an object, wait for an unload operation, wait for a load operation, and load an object. The platform may also go to other object waypoint locations to perform the load operation.

Type: Integration

Method: Test/Demonstration

Description: Create a waypoint sequence involving load and unload operations where the platform may have to travel to another location for an object, the platform may have to wait for the object, or may leave the area if the object is not there.

Results: The transportation operations were successful and the platforms were able to load/unload objects with or without waiting and with or without traveling to another location.

Recommendations/Updates: If the user wants to drop an object off and wait a certain amount of time before it comes back to be picked up, the user must create two waypoints to accomplish this task. This should be an option at a single waypoint in the future to make the operation easier to develop in a scenario.

11. SENSORS/INTELLIGENCE

There are several types of sensors represented within FOCUS. For an accurate representation, each different type of sensor has its own set of attributes and methods of scanning, collection and acquisition. As part of the V&V of the FOCUS model, each sensor's scanning, collection, and acquisition capabilities were verified and validated. Parametric sensitivities were run on appropriate algorithms in a stand-alone environment. The outputs were analyzed with the given inputs to determine if the algorithm performed how it was intended. The test cases provide a review of the coded algorithms to ensure they function as intended. The following sections state the capabilities that were verified and validated for each sensor used in the FOCUS model.

11.1 Moving Target Indicator (MTI) Radar. The MTI Radar methodology is documented in the Surveillance Radar Methodology Specification and has been validated by a Radar SME. As part of the Radar verification, the sensor's detection methodology was verified, in particular, the general Radar search and target acquisition methodology and the sensor scanning timeline.

11.1.1 Search. The MTI search methodology has two options: a simple scan pattern and a complex pattern. The simple scan pattern uses a scan rate and beam width that is moved across the azimuth limitations of the sensor. The complex scan pattern treats the entire azimuth limitations as the field of view and creates random dwell windows for any targets in view.

Test: MTI Radar Simple Search

Purpose: Examine whether the simple search pattern steps across the azimuth field of regard using the beam width, scan rate, and dwell time. Ensure that targets in the field of view are able to be detected.

Objective: Verify the field of view of the radar simple search pattern using the beam width and vertical limitations. Show that the field of view moves across the field of regard and repeats at the start. Verify that targets are correctly identified to be in the sensor field of view.

Type: Integration

Method: Test/Demonstration

Description: Setup a simple radar scenario using different beam width, scan rates, and azimuth limits. Visually see the field of view go across the field of regard and output the angles of the field of view as it is moved.

Results: The simple radar scan pattern correctly creates the field of view the size of the beam width and moves across the azimuth limits.

Test: MTI Radar Complex Search

Purpose: Test that the complex search pattern includes the entire azimuth limits and that targets are given random dwell times based on the inputs. Ensure that targets in the azimuth limits are identified as being in the field of view.

Objective: Verify the selection of random dwell times for targets and the creation of the azimuth limits field of view.

Type: Integration

Method: Test/Demonstration

Description: Create a simple radar scenario using the complex scan pattern to visually see the azimuth limited field of view and examine when targets are detected to see the random dwell times and scan times.

Results: The complex scan pattern is successfully created enables detection of targets at random dwell times.

11.1.2 Target Acquisition. The radar target acquisition methodology allows for two methods: the Golden Point method and a P(d) versus range table. The Golden Point method uses a 50% detection range and is based on the two-way radar range equation.

Test: MTI Radar Golden Point Method

Purpose: Verify that the Golden Point method give the validated probability of detection for a given sensor-target pairing.

Objective: Verify that the golden point method returns the validated probability of detection using a variety of input parameters.

Type: Unit

Method: Test

Description: Run the golden point algorithm using a variety of inputs to compare against the standalone model output.

Results: The golden point method returns the correct probability of detection given valid inputs.

11.1.3 Timing. The Radar sensor scanning timeline included analysis of the regional search time, detection time, collection time, and the pointing angle of the radar for each scan. This test examined both the simple and complex scan patterns.

Test: MTI Radar Time Considerations

Purpose: Check that the radar methodology accounts for all delays in scanning and detecting targets.

Objective: Verify the timing of all operations of the radar methodology including moving the field of view, dwell times, scan times, and detection times.

Type: Integration

Method: Test/Demonstration

Description: Using simple radar simulations using both the simple and complex scan pattern, output the timing of all radar related events to verify the timing with what is validated in the methodology.

Results: The timing of all events in the radar methodology was verified including dwell and scan times as well as detection times.

11.1.4 Probability of Detection Sensitivity. Sensitivity testing conducted on MTI radar sensor types were carried out using the Golden Point methodology. The Golden Point method uses a single radar specification reference point to compute any other $P(d)$ instance based on the radar range equation. The radar range equation is an academic accepted algorithm for quantifying the performance of radars.

Conclusions

The sensitivity results in Section 11.1.4 illustrate the impact of key parameters utilized to compute the $P(d)$ of MTI radar sensor types. The sensitivity analysis performed on these parameters provided results that are consistent with expected values of the MTI radar sensor performance methodologies.

11.1.5 Tactical Parameter Sensitivities. The MTI radar representation has been utilized in a number of different studies that show how the representation will give varied results depending on tactical parameters such as operating ranges and the number of Coherent Processing Intervals (CPI) per dwell.

The sensitivity results in Section 11.1.5 illustrate the impact of tactical parameters to MTI radar sensor representation. Alterations to these parameters provided a high level overview that was consistent with expected values which represent MTI radar sensors. These results further support the notion of varying sensor performance dependent on the tactical environment.

11.2 Synthetic Aperture Radar (SAR). The SAR sensor module was documented in the SAR Methodology Specification. This methodology was reviewed and validated by the SMEs on the AMSAA Wide Area Surveillance Team. The primary elements that were validated and verified are the strip mode search pattern, spot mode identification, the National Imaging Interpretability Rating Scale (NIIRS) acquisition methodology, and timing constraints for the

entire search, collection and image exploitation process. Below are the key facets to the SAR detection methodology that were verified.

11.2.1 Search. The SAR search pattern is a strip scan that breaks a large area into a series of strips that are systematically scanned. If a target is detected, the SAR finishes the strip then switches to a spot mode to identify the target using a higher resolution field of view.

Test: SAR Search Pattern

Purpose: Check that the SAR methodology completes the strip search using the wide field of view and switches to the narrow field of view for spot mode identifications at the end of strips.

Objective: Verify the implementation of the SAR search methodology as specified.

Type: Integration

Method: Test/Demonstration

Description: Create a simple SAR simulation that will create the strips from a large AOI. Execute the search to see the systematic strip search and spot identification. Output the events to compare to the validated specification.

Results: The SAR search methodology successfully creates the strips and searches the blocks in order. When a target is identified, the SAR finishes the strip and then goes to spot mode using the narrow field of view. The SAR then continues the strips.

11.2.2 Target Acquisition. The SAR target acquisition methodology is based on the NIIRS system to determine how a human would interpret the image to detect/ID a target.

Test: SAR NIIRS-based P(d)

Purpose: Verify that the SAR acquisition methodology produces the validated P(d).

Objective: Verify the P(d) of the SAR methodology by comparing the output with a standalone model using a variety of inputs.

Type: Unit

Method: Test

Description: Run the acquisition methods to produce the probability using a variety of inputs.

Results: The FOCUS results were consistent with the standalone model for all inputs.

11.2.3 Timing. The SAR sensor scanning timeline analyzed of the delays of collection times, switching modes, and SAR image exploitation.

Test: SAR Time Considerations

Purpose: Check that the SAR methodology follows the timing constraints as outlined in the methodology specification.

Objective: Verify the timing of events in the SAR module such as block collection times, times to switch modes, and the image exploitation time.

Type: Integration

Method: Test/Demonstration

Description: Create a simple SAR scenario that involves a strip search, detection of a target, and spot mode identification. Verify the time of events as they occur.

Results: The SAR methodology timing was consistent with the specification.

Recommendations/Updates: The image exploitation and detection of a target can vary greatly depending upon the inputs and may need further research and improvements.

11.2.4 Probability of Detection Sensitivity. Sensitivity testing conducted on SAR sensor types were carried out with the NIIRS to P(d) methodology. The NIIRS to P(d) methodology is a SME accepted algorithm based on the same imagery interpretability principles as the EO/IR Acquire methodology.

Conclusions

The sensitivity results in Section 11.2.4 illustrate the impact of key parameters utilized to compute the P(d) of SAR sensor types. The sensitivity analysis performed on these parameters provided results that are consistent with expected values of the SAR sensor performance methodologies.

11.2.5 Tactical Parameter Sensitivities. The SAR representation has been utilized in a number of different studies that show how the representation will give varied results depending on tactical parameters such collection time and exploitation time.

11.3 Electro-Optical/Infrared (EO/IR) Imaging Sensors. The detailed EO/IR methodology can be found in the EO/IR Imaging Sensor Methodology Specification. The methodology has been reviewed and validated by the SMEs on the AMSAA Target Acquisition Team. The primary elements that were validated and verified are the search pattern and footprint generation methods, the target acquisition methods including Acquire-TTPM and the Target Acquisition Draw Methodology (TADM), and the timing constraints based on the Time Limited Search Methodology.

11.3.1 Search. The search methodology is comprised of several methods integrated together including the choice of a FOV, moving the FOV across a field of regard (FOR) or in another pattern, and creation of the footprint.

Test: EO/IR FOV Choice

Purpose: Test that the EO/IR sensor implementation correctly chooses the desired FOV based on the methodology specification.

Objective: Verify that the FOV chosen for search is the widest FOV with at least 0.7 probability of detection for a standard NATO target.

Type: Unit

Method: Test

Description: Given a sensor with several FOV modes, determine which is the widest with at least 0.7 probability of detection. Run the modes through the *ChooseBestScanMode* method to verify the correct result. This test assumes that the probabilities returned have been verified in another test.

Results: The implementation chooses the correct FOV.

Test: EO/IR FOR Search

Purpose: Ensure that the EO/IR sensor implementation correctly conducts a horizontal and/or vertical field of regard search.

Objective: Verify that the EO/IR sensors are capable of performing a field of regard search that can go horizontally between a min and max angle and/or vertically between a min and max angle.

Type: Integration

Method: Test/Demonstration

Description: Given a defined FOR for a sensor with a given FOV, demonstrate in a scenario the search movement of the FOV across the FOR. Output the focal point angles to verify the incremental movement across the FOR.

Results: The implementation successfully moves the FOV across the FOR and at the end of the FOR, the FOV is moved back to the starting side.

Recommendations/Updates: An option should be added to go back and forth with the FOV in the FOR, an option to choose which direction to move (vertical first, horizontal first, serpentine, etc.).

Test: EO/IR Point Search

Purpose: Test the EO/IR sensor implementation so that it correctly sets the footprint on a defined point including a track, named area of interest (NAI), or any other object to be identified.

Objective: Verify that the sensor is capable of conducting search of a given point that is updated over time with movement of either the sensor or the object being looked at. This search type is used for point surveillance, tracking a target, and identifying a detected object.

Type: Integration

Method: Test/Demonstration

Description: Given a defined NAI, have the sensor conduct a point surveillance search, once a target is detected, have the sensor track and identify the target.

Results: The implementation successfully keeps the FOV on the point for both stationary and moving locations.

Test: EO/IR Area Search

Purpose: Verify that the EO/IR sensor implementation correctly moves the FOV through an area search pattern.

Objective: Verify that the implementation is capable of moving the footprint through an area in a defined pattern to cover the entire area. The area is defined by a set of small area blocks that are marked as scanned when the footprint is over them. The algorithm takes a “brute force” approach with the blocks to systematically cover them all.

Type: Integration

Method: Test/Demonstration

Description: Given a defined NAI, have a sensor search the area several times. Track the number of area blocks scanned versus the total number of area blocks to ensure that all the blocks are scanned.

Results: The implementation successfully scans all the area blocks.

Recommendations/Updates: The current method randomly chooses a starting point in the NAI and moves left to right and down the grid, returned to the top to finish. Another option should be added to start in the middle of the NAI and spiral out (according to new TTPs). Also, the brute force method of searching area blocks can give some random looking patterns and is memory and computationally expensive—the search pattern should have a smoother pattern.

Test: EO/IR using Fixed Sensor Mount

Purpose: Test to ensure that the sensor implementation constrains fixed sensors to the defined horizontal and vertical angles.

Objective: Verify that a sensor marked as fixed is constrained to the angles defined for all search and target acquisition tasks.

Type: Integration

Method: Test/Demonstration

Description: Given a fixed sensor type, conduct an area search. Track the horizontal and vertical angles for all tasks.

Results: The sensor stays at the constrained angles for all tasks.

11.3.2 Target Acquisition. EO/IR target acquisition performance is measured using the Acquire-TTPM model, which determines a probability of detection and identification between a sensor and a target. The Acquire-TTPM implementation for each EO/IR sensor type (IR, Television (TV), Image Intensifier (I2), and Direct View Optics (DVO)) maps back to a Night Vision Electronic Sensor Directorate (NVESD) engineering level model. The Acquire model is a rolled up version of the engineering level model that is less computationally expensive and more appropriate for real time simulations. Acquire is a validated model backed by empirical data and is the Army standard for EO/IR target acquisition performance. The TADM is used in conjunction with the Acquire output to decide whether or not a target is detected. The TADM was altered in FOCUS to not allow bad draws to have a single target or single sensor become incapable of being detected or making detections. The change is that the observer/target draw is redrawn each detection opportunity.

Test: EO/IR Acquire-TTPM

Purpose: Verify that the implementation of Acquire-TTPM produces the same results as the stand-alone Acquire-TTPM model.

Objective: Benchmark the FOCUS Acquire-TTPM implementation against the validated stand-alone Acquire-TTPM model from the Target Acquisition Team. Run the tests under a variety of conditions and examine trends to ensure verified results. Inspect code to verify key equations and data.

Type: Unit

Method: Test/Code Inspection

Description: The code inspection will consist of the following:

- *General Methods* – These are the methods used in the target acquisition process by all imaging sensors including the characteristic dimension, up slant angles, etc. The equations and methods were compared to the validated equations in the methodology description.
- *V50 Values* – The multiple calculations performed when running Acquire to attain the V50 value (the line pair criteria for a given acquisition level) of a sensor was verified. Comparisons were made to the SME approved methodology description.

- *IR Calculations* – The multiple calculations performed when running Acquire particular to IR sensors were verified. Comparisons were made to the stand-alone model as well as the SME approved methodology description.
- *Visual Calculations* – The calculations performed when running Acquire for visual (TV, I2, and DVO) sensors were examined. Comparisons were made to the stand-alone model as well as the SME approved methodology description.

The test consists of several input sets executed on the module. The sets vary multiple inputs and step through long ranges of a single input as well as include boundary conditions (setting inputs to zero).

Results: The implementation matches the equations in the methodology description and the execution results matched the stand-alone model.

Test: EO/IR TADM

Purpose: Ensure that the TADM produces the correct draw values that are compared to the probability of detection/identification.

Objective: Verify the distribution created by the TADM and the comparison of the draw to the probability produced by the target acquisition methodology used for EO/IR sensors.

Type: Unit

Method: Test

Description: Run several iterations of the draw calculations to examine the distribution created by the methodology. Compare the distributions to the expected distribution of the methodology.

Results: The distributions created matched the expected results.

Recommendations/Updates: Allow the user to select between the standard TADM and the FOCUS-altered TADM.

11.3.3 Timing. The imaging sensor timeline is a modified version of the Time Limited Search (TLS) methodology with an incremental sampling approach. The TLS method assumes that nothing changes in the FOV during the time it is being searched. Since FOCUS looks at much smaller vignettes and specific events than typical combat simulations, the sampling of the FOV must be much faster in order to avoid targets passing through the FOV during the search time. The FOCUS team developed the incremental sampling approach and had it validated by the EO/IR SME.

Test: EO/IR Time Considerations

Purpose: Examine the timing of events in the integrated EO/IR implementation to ensure it matches the guidelines set forth in the methodology description.

Objective: Verify the timing of events in the overall implementation of the EO/IR search and target acquisition process to include:

- Search time sample rate
- Time to Detect a single target
- Time to Detect additional targets in a FOV
- Delay time for switching FOV of a sensor
- Delay time for performing identification
- Delay time for identifying additional targets
- Time to stay on (search) a particular FOV (empty FOV time)

Type: Integration

Method: Test/Demonstration

Description: Set up a vignette that includes searching an empty FOV, detecting a single target in a FOV, detecting multiple targets in a FOV, and identifying multiple targets. Record the timing of events.

Results: The implementation events occur at the correct times as stated in the methodology description.

11.3.4 Probability of Detection Sensitivity. Sensitivity testing conducted on imaging sensor types were carried out using the Acquire-TTPM methodology with the P(d) given infinite time (PInf) metric. Acquire-TTPM is an Army standard standalone model that performs a curve fit to the engineering level EO/IR target acquisition performance models developed by NVESD. The NVESD models are based on empirical data from perception experiments of Soldiers finding targets in EO/IR imagery.

Conclusions

The sensitivity results in Section 11.3.4 illustrate the impact of key parameters utilized to compute the PInf using the Acquire-TTPM methodology. The sensitivity analysis performed on these parameters provided results that are consistent with expected values of the EO/IR target acquisition performance methodologies.

11.3.5 Tactical Parameter Sensitivities. The EO/IR representation has been utilized in a number of different studies that show how the representation will give varied results depending on tactical parameters such as system platform behavior, environment, FOV selection and search pattern.

11.4 SIGINT. SIGINT consists of both Electronic Intelligence (ELINT) and Communications Intelligence (COMINT). The primary elements of the SIGINT V&V are the search methods for both intelligence types, the target acquisition methods, timing considerations, and direction finding algorithm. The SIGINT methodology is a basic representation (level 1) that is documented in the SIGINT Methodology Description and validated by the appropriate SMEs.

11.4.1 Search. ELINT receivers have two options for searching, a fixed horizontal FOV and a FOV determined by a gimbal speed that scans across the horizontal plane. COMINT receivers are implemented with a full 360 degree coverage FOV.

Test: SIGINT Search

Purpose: Ensure that the ELINT implementation creates a fixed horizontal FOV given the appropriate parameters or uses a gimbal speed to determine the area scanned. Also ensure that the COMINT implementation has 360 degree coverage.

Objective: Verify the search patterns of the SIGINT receivers including the fixed FOV, the movement of the FOV using the gimbal speed, and a static 360 degree coverage area. Verify that targets are identified as being in the FOV at the correct time.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette for each type of FOV and track the point angle of the FOV for the fixed angle and gimbal speed methods. Place targets at different positions that should come into the FOV at certain times. The 360 degree coverage should be able to detect all targets in the FOV at all times.

Results: The FOVs are successfully created and targets are identified as being in the FOV at the appropriate times.

Recommendations/Updates: Update the FOVs to include look down angles. Currently all vertical angles are considered in the FOV.

11.4.2 Target Acquisition. The ELINT and COMINT methodologies both use the standard one way radar range equation with or without the system sensitivity input as the algorithm for determined the probability of detection.

Test: SIGINT Probability of Detection

Purpose: Verify that the SIGINT detection methodology (one way radar range equation) produces the correct result.

Objective: Benchmark the one way range equation against the validated stand-alone model for the one way radar range equation with and without using the system sensitivity input.

Type: Unit

Method: Test

Description: Execute the detection methodology against a set of inputs. The inputs should consider boundary conditions and the ability to look at known trends in the results. Compare the results against the stand-alone model.

Results: The results matched the stand-alone model.

Test: SIGINT Prerequisites for Detection

Purpose: Ensure that the SIGINT implementation makes the required checks to determine if an emitter is detected by a receiver.

Objective: Verify the constraints on detection such as frequency ranges, emitter type, LOS, in FOV, and comparison to random draw.

Type: Unit

Method: Test

Description: Execute the target acquisition process under a variety of conditions to test the frequency constraints, targets in the footprint, the type of emitter, LOS, and random draw.

Results: The implementation correctly assesses all conditions for detection.

11.4.3 Direction Finding. After detection, a SIGINT system will create lines of bearing (LOBs) and attempt to estimate a target's location. The process of creating a LOB and the location estimate of the target was derived from a stand-alone model that weighs the angle of intersection between LOBs and creates a weighted mean location and standard deviation.

Test: SIGINT LOB Calculation

Purpose: Test that the SIGINT methodology successfully creates LOB(s) using an LOB rate per second and angular error for input into a direction finding algorithm.

Objective: Verify the creation of LOBs using the angular error and that the correct number of LOBs is created for each detection interval.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette in which several detections occur from several receivers and examine the LOBs that are created. Output the directions for all the LOBs and the total number created for each detection.

Results: The number of LOBs and the directions of the LOBs were verified.

Recommendations/Updates: The methodology for when to combine LOBs is unclear and more options should be made available for determining when to combine into a location.

Test: SIGINT Direction Finding Algorithm

Purpose: Verify that the SIGINT methodology successfully combines the LOBs into a single location with error.

Objective: Benchmark the location and error of the FOCUS implementation against the stand-alone model.

Type: Unit

Method: Test

Description: Given a set of LOBs, execute the direction finding algorithm. Compare to the stand-alone result.

Results: The FOCUS method matched the stand-alone model.

11.4.4 Timing. The SIGINT timeline includes time for search, detecting emitters, creating LOBs and time periods for when to combine the LOBs.

Test: SIGINT Time Considerations

Purpose: Ensure that the SIGINT implementation uses the correct timing for events as outlined in the methodology description.

Objective: Verify the timing of events in the SIGINT process including the sampling interval, the time to create LOBs, and the time periods for when to combine LOBs.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute the full SIGINT search, collection, and direction finding processes. Track the timing of events and compare to methodology specifications.

Results: The timing of events matched the specification.

11.4.5 Probability of Detection Sensitivity. Sensitivity testing conducted on SIGINT sensor types were carried out with a methodology utilizing the one way range equation with System Sensitivity. The One Way Range Equation is an academic accepted algorithm for quantifying the amount of energy passed from an emitter to a receiver.

Conclusions

The sensitivity results in section 11.4.5 illustrate the impact of key parameters utilized to compute the Probability of Detection on SIGINT sensor types. The sensitivity analysis performed on these parameters provided results that are consistent with the expected values of the SIGINT performance methodologies.

11.4.6 Tactical Parameter Sensitivities. The SIGINT representation has been utilized in a number of different studies that show how the representation will give varied results depending on tactical parameters such as LOB per second.

11.5 Human Intelligence (HUMINT). The HUMINT module in FOCUS models the effects of HUMINT on a vignette and is documented in the HUMINT Methodology Description. The primary elements of the methodology are initiation of a HUMINT information transfer, making a decision on what if any information is transferred, the reporting of information to a commander, and coercion (i.e., changes to the cooperation levels between collectors and providers).

Test: HUMINT Process Flow

Purpose: Ensure that the overall process flow of HUMINT between collectors and providers, and consequently between collectors and commanders, occurs as described in the methodology description.

Objective: Verify that HUMINT interactions successfully execute with the correct time delays including:

- Initiation of the transfer when a collector detects a provider and moves to the provider
- Conduct information transfer between the collector and provider
- Transfer between the collector and commander
- Coercion
- Commander decision points

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to have a collector detect a provider, approach the provider and initiate the information transfer. Conduct the information transfer and the transfer to the commander. Execute a coercion event and repeat. Output the events as they occur.

Results: The process flow of the HUMINT collection and transfer was successful. The coercion effects were verified and the commander decision was successful. All timing was verified.

Test: HUMINT Information Transfer

Purpose: Test that the HUMINT process transfers the correct amount of information based on the probabilities of transfer defined by the user.

Objective: Verify the probability of transfer is correctly used to determine what information is passed between the collector and provider.

Type: Unit

Method: Test

Description: Conduct trials of the transfer module to confirm the distribution of information transfers according to the probabilities.

Results: The transfer rates matched the expected distributions.

11.6 Laser Designation (LD). The laser designation module in FOCUS models interactions between a laser coupled with an EO/IR camera and a guided munition. The processes include putting the spot on target, the trajectory of the guided munition, the seeker behavior of the guided munition, and the calculation of a SNR between the laser and seeker using the Night Vision Laser Designation Model (NVLaserD).

Test: LD NVLaserD Algorithm

Purpose: Ensure that the SNR computed in FOCUS match the SNR computed by the standalone NVLaserD model.

Objective: Verify the correct calculation of SNR between the laser and seeker.

Type: Unit

Method: Test

Description: Execute the SNR calculation for several sets of inputs and compare to the results of the standalone model.

Results: The SNR calculated matched the standalone model.

Test: LD Guided Munition Trajectory

Purpose: Ensure that the computed trajectory used by the guided munition representation is comparable to the known trajectory data.

Objective: Verify that the dynamic trajectories in the model are comparable to the known trajectory data.

Type: Integration

Method: Test/Demonstration

Description: Run scenarios at the ranges used in the known trajectories and compare the resulting trajectories to the known points.

Results: The computed trajectories were comparable to the trajectory data with an acceptable amount of error.

Test: LD Guided Munition Velocity

Purpose: Verify that the computed velocity for a given time from launch matches the guided munition data provided for the implementation.

Objective: Verify that the velocity for a given time is computed correctly and the overall velocity curve match the provided data.

Type: Unit

Method: Test

Description: Create a velocity curve over time from the FOCUS implementation and match to the known velocity curve.

Results: The velocity curve matched the provided velocity data.

Test: LD Guided Munition Seeker Search

Purpose: Verify that the guided munition seeker uses the correct search pattern for finding the laser. Also ensure that the footprint correctly identifies the laser as being in the FOV when appropriate.

Objective: Verify the search pattern and footprint check of the seeker. The seeker FOV is an eight degree circle that moves in a rectangular pattern when searching and stays on the laser spot once it is found.

Type: Integration

Method: Test/Demonstration

Description: Create a simple LD scenario to check the search pattern and the footprint check. Ensure that the pattern matches the rectangular specification and the footprint is the eight degree circle. Ensure that the laser is not found until it is in the FOV.

Results: The search pattern matched the specification and the eight degree circular FOV correctly identified when the laser spot was in view.

Test: LD Time Considerations

Purpose: Test that the LD methodology follows the timing rules in the specification for incrementing movement, searching with the seeker, etc.

Objective: Verify the time of events associated with the LD methodology.

Type: Integration

Method: Test/Demonstration

Description: Create a simple LD scenario that will test all processes of the LD methodology. Record the timing of events and compare to the expecting time delays.

Results: The timing of all events in the LD methodology was verified.

12. COMMUNICATIONS

12.1 Communications. When communication takes place between a sensor and the Command and Control (C2) node, there is a message delay that is calculated. In addition to the message delay, a message completion rate is also used.

Test: Messaging

Purpose: Ensure that messages from sensors/platforms to a C2 node are delayed with the appropriate time and that the completion rate match the desired distribution.

Objective: Verify the use of delay times and completion rate in the messaging module.

Type: Unit

Method: Test

Description: Run several iterations of the send message module to calculate the delay times assigned and the completion distribution. Compare to expected values.

Results: Delay times and completion rates matched the expected values.

12.2 RF Propagation. The FOCUS model supports an architecture that allows the RF loss to be calculated between two points. It uses a hybrid method that utilizes Okumura-Hata equations and the Terrain Integrated Rough Earth Model (TIREM). The premise of this method is that when there are no LOS blocks due to buildings, TIREM is used. Otherwise a weighted-Hata method is used. The weighted-Hata method weights the path loss of each type of terrain type (open, med-city, large-city, etc.) with the ratios of the path in each of the types of terrain. If one of the platforms is airborne and/or above the buildings, a single-knife-edge approximation is used to calculate the RF loss. There are also environmental attenuations due to factors such as rain, fog, and vegetation.

12.2.1 TIREM. As part of the V&V process of the FOCUS model, TIREM will be compared to several test conditions provided in the TIREM documentation.

Test: RF Propagation using TIREM

Purpose: Verify that the TIREM model matches the validated test data from the TIREM documentation.

Objective: Benchmark the TIREM results against the validated stand-alone data.

Type: Unit

Method: Test

Description: Execute the TIREM model with the supplied test run inputs from the TIREM documentation and compare results.

Results: The FOCUS TIREM implementation matches the expected results.

12.2.2 Hata Model. The Okumura-Hata equations will be compared to values generated from the Okumura-Hata equations in an Excel spreadsheet (stand-alone model).

Test: RF Propagation using HATA

Purpose: Verify that the HATA model matches the validated test data from the HATA stand-alone model.

Objective: Benchmark the HATA results against the validated stand-alone data.

Type: Unit

Method: Test

Description: Execute the HATA model with various test run inputs and compare results to the HATA spreadsheet model.

Results: The FOCUS HATA implementation matches the expected results.

12.2.3 FOCUS Hybrid Method. The hybrid method is tested using collected data from Boulder, CO and Rosslyn, VA. The RMS error of the FOCUS hybrid method must be lower than the root-mean-square (RMS) error when using the Vertical Plane Launch (VPL) model.

Test: RF Propagation using the FOCUS Hybrid Method

Purpose: Verify that the FOCUS Hybrid method performs better than the VPL model results.

Objective: Benchmark the Hybrid method results against the validated stand-alone data.

Type: Unit

Method: Test

Description: Execute the Hybrid model with test data inputs and compare results to the VPL model.

Results: The FOCUS Hybrid method error was lower than the VPL model.

13. MISSIONS AND BEHAVIORS

13.1 Mission Flow Architecture. The mission flow architecture is a tree-like structure that allows the user to setup a sequence of automated events to script a scenario. When creating a scenario, the user is able to create mission, conditional, and event objects and add these objects to the mission flow as nodes in the architecture. When the scenario is run, the nodes are running according to the rules of the architecture as defined in the user's manual. The execution of these rules was verified.

Test: Mission Flow Execution

Purpose: Ensure that the mission flow architecture successfully executes the user defined nodes following the rules outlined in the user's manual.

Objective: Verify the execution and timing of the mission flow architecture.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette that will exercise the rule set for the mission flow architecture involving missions, conditions, events, different priorities, child missions, etc.

Results: The mission flow architecture executes the nodes as expected.

Recommendations/Updates: There are numerous updates that could be made to the interface for creating the mission flow to make the setup easier and the product look cleaner. A large number of nodes and connections make the interface difficult to understand.

13.2 Mission Types. Missions are sets of actions that define an automated process for completing a complex behavior. Each mission has a set of rules for what assets can perform the task, what actions are performed over time, and how the mission can end. Each mission type was verified for correct implementation.

Test: Asset Assignment

Purpose: Ensure that each mission type allows the correct and best suited asset(s) to be assigned to it.

Objective: Verify that only the best suited and appropriate asset(s) is allocated to each mission type.

Type: Unit

Method: Test

Description: Create a unit test that executes the assignment of the mission to the assets and ensure that the known most appropriate asset is allocated to the mission. Vary the asset pool with different priorities, sensor types, and distances from objective.

Results: Each mission successfully allocates the best asset for the mission.

13.2.1 Cover AOI. The Cover AOI mission is intended for platforms with EO/IR sensors and defines the automated process for searching an area of interest. The Cover AOI allows completion after one coverage cycle or continuous coverage.

Test: Cover AOI Mission Execution

Purpose: Test that the cover AOI mission executes the required processes in the correct order and at the correct time and that it ends the mission when appropriate.

Objective: Verify the execution order and time of all mission actions and the completion of the mission if it is defined to end after one coverage cycle of the area.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute a cover AOI mission and use the Debug output files to track the execution of the actions in the mission. Execute a mission using one coverage cycle completion option and the continuous coverage option. Execute the mission using varied number of assets completing the task.

Results: The cover AOI mission successfully executes the single coverage cycle and continuous coverage options. The mission is capable of using one or more assets to cover an area by dividing the area into smaller chunks.

13.2.2 Identify Track. This mission can be assigned to a platform with a sensor that can ID a tracked target (typically EO/IR, but SAR is also viable). This mission continues to execute until the unidentified track is identified.

Test: ID Track Mission Execution

Purpose: Ensure that the ID track mission executes the required processes in the correct order and at the correct time and that the mission ends when appropriate.

Objective: Verify the execution order and time of all mission actions and the completion of the mission when the track is identified.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute an ID track mission and use the Debug output files to track the execution of the actions in the mission.

Results: The ID track mission successfully executes the actions required. When the target is identified, the mission is completed.

13.2.3 Move. The move mission assigns one or more platforms to move on a set of waypoints or to a selected location.

Test: Move Mission Execution

Purpose: Verify that the move mission executes the required processes in the correct order and at the correct time and that the mission ends when appropriate.

Objective: Verify the execution order and time of all mission actions and the completion of the mission when the objects reach the end of the waypoint list or to the specified location.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute a move mission and use the Debug output files to track the execution of the actions in the mission. Execute the mission with one and many assets with both the waypoint option and the selected location option.

Results: The move mission successfully executes for one and many assets as well as for the waypoint movement and selected location options.

13.2.4 MTI Coverage. The MTI coverage mission is the automated behavior for providing coverage of an area using surveillance radar.

Test: MTI Coverage Mission Execution

Purpose: Ensure that the MTI coverage mission executes the required processes in the correct order and at the correct time.

Objective: Verify the execution order and time of all mission actions.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute an MTI coverage mission and use the Debug output files to track the execution of the actions in the mission.

Results: The MTI coverage mission executes successfully. The mission is continuous unless the platform/sensor is assigned to a higher priority mission.

13.2.5 SIGINT Coverage. The SIGINT coverage mission is the automated behavior for providing coverage of an area using a SIGINT receiver.

Test: SIGINT Coverage Mission Execution

Purpose: Verify that the SIGINT coverage mission executes the required processes in the correct order and at the correct time.

Objective: Verify the execution order and time of all mission actions.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute an SIGINT coverage mission and use the Debug output files to track the execution of the actions in the mission.

Results: The SIGINT coverage mission executes successfully. The mission is continuous unless the platform/sensor is assigned to a higher priority mission.

13.2.6 Track Target. The track target mission defines the behavior for following (tracking) a target with an EO/IR sensor.

Test: Track Target Mission Execution

Purpose: Ensure that the track target mission executes the required processes in the correct order and at the correct time.

Objective: Verify the execution order and time of all mission actions.

Type: Integration

Method: Test/ Demonstration

Description: Create a vignette to execute a track target mission and use the Debug output files to track the execution of the actions in the mission

Results: The track target mission successfully executes.

13.2.7 SAR Coverage. The SAR coverage mission is the automated behavior for providing coverage of an area using a SAR receiver.

Test: SAR Coverage Mission Execution

Purpose: Ensure that the SAR coverage mission executes the required processes in the correct order and at the correct time.

Objective: Verify the execution order and time of all mission actions.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette to execute an SAR coverage mission and use the Debug output files to track the execution of the actions in the mission.

Results: The SAR coverage mission executes successfully. The mission is continuous unless the platform/sensor is assigned to a higher priority mission.

13.2.8 Find Track. The find track mission creates a search area based on the last known location of a target. The mission continues until the track is found. Each time the area box is scanned and the target is not found, the area is expanded.

Test: Find Track Mission Execution

Purpose: Verify that the find track mission executes the required processes in the correct order and at the correct time as well as appropriately ends when the track is found.

Objective: Verify the execution order and time of all mission actions and that the mission ends when the track is found.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette that executes the find track mission and use the Debug output files to track the execution of the actions in the mission. Create different runs to exercise finding the track and completing the mission as well as not being able to find the track and having the area expand.

Results: The find track mission successfully executes the actions including expanding the search area when the track is not found after searching the full area. The mission successfully ends when the track is found.

Recommendations/Updates: This mission should be improved to include parameters for creating the search area and how many times to expand area.

13.2.9 Collect HUMINT. There are two HUMINT collection missions, one that defines the movement of a HUMINT collector on a set of waypoints where there may be providers in the area and one that defines the HUMINT collection process given the collector and provider. The Collect HUMINT mission is a continuous process that does not end unless the collector moves to the end of a waypoint list. The Collect HUMINT from Track mission ends once the information is passed.

Test: Collect HUMINT Mission Execution

Purpose: Ensure that the Collect HUMINT missions execute the required processes in the correct order and at the correct time as well as completes the mission when appropriate.

Objective: Verify the execution order and time of all mission actions and that the mission ends when it is defined to end.

Type: Integration

Method: Test/Demonstration

Description: Create vignettes to exercise the Collect HUMINT missions, one where the collector moves on the path and collects from providers that are within the collection range and another where the provider is given to the collector. Vary other parameters such as report immediately and distance to provider.

Results: The HUMINT missions both execute successfully. The distance to provider is accurately used and the information is reported immediately when defined.

13.2.10 Convoy Support. The convoy support mission defines a behavior for coverage of an area ahead of a defined convoy object for an EO/IR sensor. The mission ends when the convoy reaches the end of its waypoint list.

Test: Convoy Support Mission Execution

Purpose: Verify that the convoy support mission executes the required processes in the correct order and at the correct time as well as completes the mission when the convoy reaches the end of the waypoints.

Objective: Verify the execution order and time of all mission actions and that the mission ends when it is defined to end.

Type: Integration

Method: Test/Demonstration

Description: Create a vignette that exercises the convoy support mission. Vary the distance between the convoy and ISR asset.

Results: The convoy support mission executes successfully using the defined distance.

13.3 Conditions. Conditions are nodes in the mission flow that can change the execution path based on the state of the simulation. Each condition was verified to make the correct decision and pass execution on the right path. The following are the condition types:

- **New Track Detected** - The new track detected condition passes the execution to the connected nodes when a new track has been created, regardless of who created the track.

- Time Condition – The time condition passes execution after it has been executed for a defined amount of time, either elapsed time or specific simulation time.
- Track Filter – The track filter takes in an existing track and checks it against one or more filters such as platform type, sensor type, located in AOI...Execution and the track are passed based on whether or not the track met the criteria selected in the filters.
- Random – The random condition passes execution based on a comparison between a random uniform draw and a selected threshold.
- HUMINT – The HUMINT condition passes execution based on whether the commander's intelligence requirements have been collected or not.
- Message Link – the message link condition passes execution when a message link is activated from the fusion flow architecture.

Test: Condition Check Execution

Purpose: Ensure that a condition passes execution to the correct path.

Objective: Verify the check method of each condition to ensure the correct passing of execution.

Type: Unit

Method: Test

Description: Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.

Results: All conditions passed execution to the correct path given the state of the simulation at the time.

13.4 Events. Events are single purpose processes that change the state of the simulation. Each event was verified to complete the action that it defines. Events include:

- Start Movement – starts the movement of a stopped platform.
- Stop Movement – stops the movement of a moving platform, can include an automatic start.
- Scan Time – turns a sensor on or off with time options.
- Activate IED – aggregates a collection of components into an IED object.
- Coercion – changes the relationship values in the HUMINT processes.
- Create track – automatically creates a track object that can be used for track missions.

Test: Mission Event Execution

Purpose: Verify that each mission event completes the action it is defined to execute.

Objective: Verify the correct execution of the mission events.

Type: Integration

Method: Test/Demonstration

Description: Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.

Results: All the events successfully execute their respective actions.

13.5 Mission Behaviors. Mission behaviors play an important role in the actions defined by missions. The behaviors dictate the movement of assets assigned to missions. Each behavior was verified to create the correct set of waypoints and assignment to the asset.

Behaviors include:

- Laser Designation Circle – creates a circular path using the platform turning radius at the safe range/angle for laser designating
- Laser Designation Hover – creates a hover point at the safe range/angle for laser designating
- Circle around Perimeter – creates a circular path centered on an AOI or other defined point with a user defined radius
- Fixed Sensor – creates a circular path centered on a defined point based on the fixed sensor constraints
- Follow Waypoint List – sets the asset to follow a defined waypoint list
- Hover Circle – creates a circular set of waypoints where the asset will stop and hover at points with LOS to a defined center point
- Hover Keep LOS – creates a point to hover at based on LOS to a defined location
- Hover/Park – creates a point to hover at a defined location
- Keep LOS – creates an orbit based on the turning radius at a location with LOS to a defined target location
- Move – creates waypoints to a location such as the center of an AOI, a landing point, or a launch point
- Move to Track – creates waypoints to a track location
- Moving Circle – creates a series of orbits along a path
- Perpendicular – creates a long oval path perpendicular to a target location
- Remain at Location – sets a platform to stay at the current location
- Remain at Standoff – sets an orbit based on the turning radius at a defined standoff from another location
- Serpentine – creates a serpentine “s” pattern over an AOI

Test: Constructing Behaviors

Purpose: Ensure that the behaviors are constructed according to how they were defined.

Objective: Verify that each behavior is constructed how it is defined using the user parameters and that the mission assets are successfully assigned to the waypoints created.

Type: Integration

Method: Test/Demonstration

Description: Create vignettes to exercise each mission behavior using a variety of parameters.

Results: All the mission behaviors are constructed according to their definition using the user parameters and assets are successfully assigned to follow the path.

14. ANALYSIS TOOLKIT

The analysis toolkit was verified to ensure that it successfully reads in and converts scenario output data to data traces. Because the analysis toolkit is embedded into FOCUS, the same methodology used to read in the main simulation file is also used to read in the scenario output file; thus, this part of the package did not require verification. The important features of the analysis toolkit that were verified include:

- Raw Data Type Generation
- Operators
- Mapping

14.1 Raw Data Types. Raw Data Types are generated directly from processing the FOCUS output file. Typically, the raw data does not involve any type of operation and is simply the organization of data into a specific data “trace,” such as the number of detections over time organized by sensor. The data traces are constructed into a tree structure defined by the data filters chosen by the user. Each raw data type has specific filters that are applicable such as sensor type, detecting platform, and target platform for the number of detections. The reading/processing of records, the extraction of data values, and the filtering of data were the main processes verified.

Test: Analysis Record Processing

Purpose: Ensure that each record type is processed correctly and used for the correct raw data trace.

Objective: Verify the data read from the record types is correct and that the appropriate records are matched to the raw data types.

Type: Unit

Method: Test

Description: Create a unit test to read a set of known records and match them to a set of known raw data types. Compare the executed results to the expected results.

Results: Each record type is correctly identified and matched to the appropriate raw data types.

Test: Analysis Raw Data Trace Processing

Purpose: Ensure that each raw data type extracts the correct data values from the output file.

Objective: Verify the data trace values produced by each raw data type.

Type: Unit

Method: Test

Description: Given an output file, create a trace for each raw data type. Compare the resulting trace to the expected result.

Results: Each raw data type pulled the correct data from the output file and processed it into the correct trace values.

Test: Analysis Data Filtering

Purpose: Ensure that each data filter correctly identifies records that pass the criteria defined for the filter.

Objective: Verify that each data filter type compares the correct values in a record to the filter criteria and makes the correct pass/fail call.

Type: Unit

Method: Test

Description: Create a set of records and filters do conduct comparisons. Compare the expected result to the executed results of each filter checked. Run for all expected filter and record pairings.

Results: All filter and record pairings successfully choose the correct pass/fail outcome.

Test: Analysis Trace Building

Purpose: Ensure that the data traces are correctly constructed using the filter levels set by the user.

Objective: Verify that the overall trace “tree” is organized using the filters in the orders set by the user. Each filter is a level in the tree.

Type: Integration

Method: Test/Demonstration

Description: Create a data trace to be created with a set of known filters. Compare the trace levels to the expected result. Vary the number of levels and filters used.

Results: The data traces were successfully created using the appropriate filter levels.

14.2 Operators. The analysis toolkit allows the user to create additional data based on operations performed on already created data elements. Each operator was verified to ensure that it produced the correct result for each single calculation. In addition to this, complex data operations that involved two data elements or lists of traces were verified to produce the correct result. The binary operators take two data elements and match the correct data points to evaluate the operator. The trace list operators take a list of traces and combine all of the traces into a final result such as taking the mean over all traces.

Test: Analysis Operators

Purpose: Ensure that each data trace operator produces the correct data value, given the corresponding inputs.

Objective: Verify the data trace operator's base purpose given a set of input data.

Type: Unit

Method: Test

Description: For each operator, execute the operation on a set of known inputs and compare to the expected outcome.

Results: All operators produced the expected result.

Test: Analysis Operators Integration

Purpose: Verify that operations on a single trace, trace lists, and two data element operators produce the correct result by pairing the correct data values to perform the operation and organize the new data trace successfully.

Objective: Verify the building of new traces according to the operator type and the scheme chosen by the user.

Type: Unit

Method: Test

Description: Create a set of data elements on which operations will be performed to exercise the use of single trace, trace lists, and binary data element operators.

- Single Trace Operator – uses a single trace to develop the new trace based on an operator such as mean value or maximum value of the trace, typically generates a single value result
- Trace Lists Operator – uses a list of traces to develop the new trace based on an operator such as mean value over time, maximum value over time, typically generates a trace over time for each base level list
- Binary Data Element Operator – uses two data elements or a data element and another value to produce a new trace based on the operator such as adding two elements, finding the distance between two vector locations, typically generates a trace with the same structure as the input

Results: Each operator type is integrated correctly to produce the expected trace(s) based the scheme selected by the user.

14.3 Data Export. The analysis toolkit is capable of exporting the data elements to a comma separated value file or Excel Spreadsheet.

Test: Analysis Data Export

Purpose: Ensure that the analysis toolkit properly exports the data elements created by the user.

Objective: Verify that the correct data is export to the file with the correct labeling.

Type: Integration

Method: Test/Demonstration

Description: Create an analysis session with known data values, export the data using both methods, and compare the results to the expected output.

Result: The toolkit successfully exports the data to a spreadsheet file.

14.4 Analysis Mapping. Another significant feature of the analysis toolkit provided to the user is the ability to map resulting data from a scenario onto an image of a vignette's operational environment. There are two forms of data mapping provided by the analysis toolkit, these two forms include coverage mapping and density mapping.

Test: Analysis Coverage Mapping

Purpose: Ensure that the analysis toolkit allows the user to extract the coverage data and maps it on the terrain image.

Objective: Verify the correct extraction of coverage data and the mapping of the data to a visualization of the terrain.

Type: Integration

Method: Demonstration

Description: Create several output files containing coverage data with known coverage locations. Use the toolkit to extract the data and use the interface to map the data on the terrain. Compare to the expected result.

Result: The coverage mapping interface has been demonstrated to produce the coverage images with the correct coverage data.

Test: Analysis Density Mapping

Purpose: Ensure that the analysis toolkit allows the user to extract vector location data and maps it on to the terrain image.

Objective: Verify the display of vector data on the density mapping interface.

Type: Integration

Method: Demonstration

Description: Create an output file with known locations of interest. Plot the vector data using the interface and compare to the expected result.

Result: The density mapping interface corrected plotted the vector data.

15. OTHER SIMULATION TYPES.

15.1 Sensor Coverage Analysis. The sensor coverage analysis simulation adds the coverage output to the main simulation. The coverage output is a grid of points over the terrain that identifies the number of times each point was covered (with LOS) by each sensor. The coverage output can then be processed in the analysis toolkit.

Test: Simulation Coverage Analysis

Purpose: Ensure that the coverage simulation records the correct coverage data for each sensor.

Objective: Verify that the coverage simulation processes each FOV investigated by the sensors to include LOS and records which points on the terrain were covered.

Type: Integration

Method: Demonstration

Description: Create a series of coverage simulations to examine the changes in the output based on the different coverage behaviors. Vary the behaviors from simple (only a few areas looked at) to complex (several areas looked at several times). Compare the resulting coverage images to the expected coverage.

Results: The coverage simulation successfully creates the coverage output and matches the expected coverage areas.

15.2 Terrain Analysis (Line of Sight and Probability of Acquisition). The terrain analysis simulation runs specific methods within FOCUS to generate line of sight and probability of detect/ID metrics over a given terrain for a set of sensor locations. For each sensor location chosen, the line of sight and probability of detecting or identifying a target is generated for each point on the terrain at a given resolution.

Test: Simulation Terrain Analysis

Purpose: ensure that the terrain simulation integrates the correct methods to produce the line of sight and probability metrics over the terrain space and sensor locations.

Objective: Verify the generation of the line of sight and probability metrics/images for a given set of sensor locations and terrain points.

Type: Integration

Method: Demonstration

Description: Create a series of terrain simulation to examine the changes in the output based on the terrain features and sensor locations. Vary the features from simple (a single building) to complex (many buildings/mountains) and vary the number of sensor locations. Compare the images/metrics to the expected patterns.

Results: The terrain simulation produced the expected metrics/images.

16. TEST CASE TABLE

The following table is an overview of all the test cases that were performed for the FOCUS V&V:

Table 2. Test Case Description and Results

Category	Test Item	Type	Method	Description	Result
Simulation Architecture	Runtime Speed Efficiency	Integration	Demonstration	A simple scenario was constructed involving platforms with sensors performing missions to search for targets. The number of platforms/sensors was increased to see the impact on run time with graphics on as well as off.	Pass
	Terrain Size	Unit	Test/ Demonstration	Load progressively larger terrains until an error occurs (memory overload).	Pass
	Number of AOIs	Unit	Test/ Demonstration	Test creating a large number of AOIs and record the effect on memory.	Pass
	Number of Buildings	Integration	Test/ Demonstration	Test creating a large number of buildings and record the effect on memory.	Pass
	Number of Platforms/Sensors	Integration	Test/ Demonstration	Test creating a large number of platforms with sensor and record the effect on memory.	Pass
	Number of Missions	Integration	Test/ Demonstration	Create a series of scenarios that increase the number of entities to see memory overload handling (if it occurs) and the general responsiveness of the software when dealing with the increased memory usage.	Pass
	Event Management	Unit	Test	Post a series of events to the event manager with varying fire times and ensure that they are sorted into the correct sequence.	Pass
	Internal Clock	Unit	Test	Post a series of events at varying times, execute the model using varying multiples of real time, and examine the real time execution times.	Pass
	Simulation Run & Iterations	Integration	Demonstration	Run a series of the scenarios with different replication numbers, times, and simulation variables.	Pass
	Action Event Management	Integration	Test/ Demonstration	Run a series of scenarios that use actions to complete behaviors. Output the times that actions are created and finished and how the parent/child actions interact.	Pass
	Scenario File I/O	Integration	Test/ Demonstration	The scenario file was tested by setting variables and saving the file, then loading the file and checking the value of the variables.	Pass
	Terrain File I/O	Integration	Test/ Demonstration	The terrain file was tested by setting variables and saving the file, then loading the file and checking the value of the variables.	Pass

Category	Test Item	Type	Method	Description	Result
Simulation Architecture	Output File I/O	Integration	Test/ Demonstration	The output file was tested by setting variables and saving the file, then loading the file and checking the value of the variables.	Pass
	Library File I/O	Integration	Test/ Demonstration	All files were tested by setting variables and saving the file, then loading the file and checking the value of the variables.	Pass
	GUI	Integration	Test/ Demonstration	Each GUI component was examined and executed for functional testing. Components with data entry were given erroneous input to test for error handling.	Pass
Environment	DTED Import	Integration	Test/ Demonstration	Each test will use a known location to compare to what is imported by the model. The image produced by the selected regions should match.	Pass
	Coordinate Conversion	Unit	Test	Exercise the coordinate conversion algorithm on a known set of data to verify the conversion from one system to another. Compare a terrain set with data generated by the geospatial SMEs for consistency.	Pass
	LiDAR Terrain Data	Unit/ Integration	Test/ Demonstration	Create a simple text file in the LiDAR format with known data and import into FOCUS. Compare the terrain elevation data structure to the known data. Using the GUI, select and compare regions that can be compared visually for correctness.	Pass
	Conversion to Indexed Array	Unit	Test	Create the transformation matrix for a given terrain and recall the index values for the Cartesian coordinates. Compare the values generated by FOCUS to those that would be found by the transformation algorithm found in literature.	Pass
	Buildings Merge	Unit	Test/ Demonstration	Create buildings with known size and locations and merge the data into the terrain. Compare the new data with the known data.	Pass
	Terrain Features	Unit	Test	Create terrain feature with a known size and location and set a ray through the feature with no start and end points. Verify the hit point and distance through the feature.	Pass
	Line of Sight	Unit	Test	Create a terrain with known obstruction points and set two points to check the space between. Change the resolution and points to check different conditions.	Pass
	Diurnal Cycle	Unit	Test	Create a diurnal cycle parameter with known values over time and query the function to get the correct value at a given time.	Pass
	Extinction Coefficient	Unit	Test	Exercise the extinction coefficient methods using the different methods to return a known value for a range and waveband combination.	Pass
	Target Acquisition Decay through Vegetation	Unit	Test	Query the database with various inputs and return known values. Calculate the degradation factor and compare to known values.	Pass

Category	Test Item	Type	Method	Description	Result
Entity Movement and Waypoints	Ground Platform Movement	Unit	Test/ Demonstration	Calculate the movement locations of platforms using various speeds, locations, and turning radii. Compare the resulting movement locations against the maximum speed and turning radius to ensure that the movement was within the threshold. Create a sequence of turns to ensure that the platforms can make every possible type of movement in all quadrants.	Pass
	Air Platform Movement	Unit	Test/ Demonstration	Calculate the movement locations of platforms using various speeds, locations, turning radii, and climb rates. Compare the resulting movement locations against the maximum speed and turning radius to ensure that the movement was within the threshold. Create a sequence of turns to ensure that the platforms can make every possible type of movement in all quadrants.	Pass
	Organization Element Movement	Integration	Test/ Demonstration	Create the formations and measure the distance between the platforms that make up the formation. Demonstrate the formation and movement along the path in a simple scenario.	Pass
	Waypoint Graph Management	Integration	Test/ Demonstration	Create a waypoint graph with different decision points and demonstrate that the object can traverse the graph using the known path. Use various speeds and turning radii to test the ability to reach a waypoint in order to move from point to point.	Pass
	Waypoint Pattern Creation	Unit	Test/ Demonstration	Create waypoint patterns using a variety of input parameters and compare the waypoint locations to see if they match the inputs. Visually inspect the pattern that is formed.	Pass
	Waypoint Payload Actions	Integration	Test/ Demonstration	Create payload actions with a variety of inputs along a sequence of waypoints to how the payload behaviors are updated as the object reaches each waypoint.	Pass
	Waypoint Entity Transportation	Integration	Test/ Demonstration	Create a waypoint sequence involving load and unload operations where the platform may have to travel to another location for an object, the platform may have to wait for the object, or may leave the area if the object is not there.	Pass
Sensors and Intelligence	MTI Simple Search Methodology	Integration	Test/ Demonstration	Setup a simple radar scenario using different beam width, scan rates, and azimuth limits. Visually see the field of view go across the field of regard and output the angles of the field of view as it is moved.	Pass
	MTI Complex Search Methodology	Integration	Test/ Demonstration	Create a simple radar scenario using the complex scan pattern to visually see the azimuth limited field of view and examine when targets are detected to see the random dwell times and scan times.	Pass
	MTI Golden Point Detection Methodology	Unit	Test	Run the golden point algorithm using a variety of inputs to compare against the standalone model output.	Pass

Category	Test Item	Type	Method	Description	Result
Sensors and Intelligence	MTI Time Considerations	Integration	Test/ Demonstration	Using simple radar simulations using both the simple and complex scan pattern, output the timing of all radar related events to verify the timing with what is validated in the methodology.	Pass
	SAR Search Methodology	Integration	Test/ Demonstration	Create a simple SAR simulation that will create the strips from a large AOI. Execute the search to see the systematic strip search and spot identification. Output the events to compare to the validated specification.	Pass
	SAR Target Acquisition Methodology	Unit	Test	Run the acquisition methods to produce the probability using a variety of inputs.	Pass
	SAR Time Considerations	Integration	Test/ Demonstration	Create a simple SAR scenario that involves a strip search, detection of a target, and spot mode identification. Verify the time of events as they occur.	Pass
	EO/IR Field of View Selection	Unit	Test	Given a sensor with several FOV modes, determine which is the widest with at least 0.7 probability of detection. Run the modes through the <i>ChooseBestScanMode</i> method to verify the correct result. This test assumes that the probabilities returned have been verified in another test.	Pass
	EO/IR Field of Regard Search Methodology	Integration	Test/ Demonstration	Given a defined FOR for a sensor with a given FOV, demonstrate in a scenario the search movement of the FOV across the FOR. Output the focal point angles to verify the incremental movement across the FOR.	Pass
	EO/IR Point/Tracking Search	Integration	Test/ Demonstration	Given a defined NAI, have the sensor conduct a point surveillance search, once a target is detected, have the sensor track and identify the target.	Pass
	EO/IR Area Search	Integration	Test/ Demonstration	Given a defined NAI, have a sensor search the area several times. Track the number of area blocks scanned versus the total number of area blocks to ensure that all the blocks are scanned.	Pass
	EO/IR Fixed Sensor Search	Integration	Test/ Demonstration	Given a fixed sensor type, conduct an area search. Track the horizontal and vertical angles for all tasks.	Pass
	EO/IR Acquire-TTPM Methodology	Unit	Test/ Inspection	The test consists of several input sets executed on the module. The sets vary multiple inputs and step through long ranges of a single input as well as include boundary conditions (setting inputs to zero).	Pass
	EO/IR Target Acquisition Draw Methodology	Unit	Test	Run several iterations of the draw calculations to examine the distribution created by the methodology. Compare the distributions to the expected distribution of the methodology.	Pass

Category	Test Item	Type	Method	Description	Result
Sensors and Intelligence	EO/IR Time Considerations	Integration	Test/ Demonstration	Set up a vignette that includes searching an empty FOV, detecting a single target in a FOV, detecting multiple targets in a FOV, and identifying multiple targets. Record the timing of events.	Pass
	SIGINT Search Methodology	Integration	Test/ Demonstration	Create a vignette for each type of FOV and track the point angle of the FOV for the fixed angle and gimbaled speed methods. Place targets at different positions that should come into the FOV at certain times. The 360 degree coverage should be able to detect all targets in the FOV at all times.	Pass
	SIGINT One Way Range Equation w/ System Sensitivity Methodology	Unit	Test	Execute the detection methodology against a set of inputs. The inputs should consider boundary conditions and the ability to look at known trends in the results. Compare the results against the stand-alone model.	Pass
	SIGINT Target Acquisition Prerequisites	Unit	Test	Execute the target acquisition process under a variety of conditions to test the frequency constraints, targets in the footprint, the type of emitter, LOS, and random draw.	Pass
	SIGINT Line of Bearing Creation	Integration	Test/ Demonstration	Create a vignette in which several detections occur from several receivers and examine the LOBs that are created. Output the directions for all the LOBs and the total number created for each detection.	Pass
	SIGINT Direction Finding Methodology	Unit	Test	Given a set of LOBs, execute the direction finding algorithm. Compare to the stand-alone result.	Pass
	SIGINT Time Considerations	Integration	Test/ Demonstration	Create a vignette to execute the full SIGINT search, collection, and direction finding processes. Track the timing of events and compare to methodology specifications.	Pass
	HUMINT Process Methodology	Integration	Test/ Demonstration	Create a vignette to have a collector detect a provider, approach the provider and initiate the information transfer. Conduct the information transfer and the transfer to the commander. Execute a coercion event and repeat. Output the events as they occur.	Pass
	HUMINT Information Transfer	Unit	Test	Conduct trials of the transfer module to confirm the distribution of information transfers according to the probabilities.	Pass
	Laser Designation NVLaserD Methodology	Unit	Test	Execute the SNR calculation for several sets of inputs and compare to the results of the standalone model.	Pass
	Guided Munition Trajectory	Integration	Test/ Demonstration	Run scenarios at the ranges used in the known trajectories and compare the resulting trajectories to the known points.	Pass

Category	Test Item	Type	Method	Description	Result
Sensors and Intelligence	Guided Munition Velocity	Unit	Test	Create a velocity curve over time from the FOCUS implementation and match to the known velocity curve.	Pass
	Guided Munition Seeker Search Method	Integration	Test/ Demonstration	Create a simple LD scenario to check the search pattern and the footprint check. Ensure that the pattern matches the rectangular specification and the footprint is the eight degree circle. Ensure that the laser is not found until it is in the FOV.	Pass
	Laser Designation Process Time Considerations	Integration	Test/ Demonstration	Create a simple LD scenario that will test all processes of the LD methodology. Record the timing of events and compare to the expecting time delays.	Pass
Communications and RF Propagation	Communications Message Completion Rates	Unit	Test	Run several iterations of the send message module to see the delay times assigned and the completion distribution. Compare to expected values.	Pass
	TIREM RF Loss Methodology	Unit	Test	Execute the TIREM model with the supplied test run inputs from the TIREM documentation and compare results.	Pass
	HATA RF Loss Methodology	Unit	Test	Execute the HATA model with various test run inputs and compare results to the HATA spreadsheet model.	Pass
	Hybrid RF Loss Methodology	Unit	Test	Execute the Hybrid model with test data inputs and compare results to the VPL model.	Pass
Missions and Behaviors	Mission Process Flow	Integration	Test/ Demonstration	Create a vignette that will exercise the rule set for the mission flow architecture involving missions, conditions, events, different priorities, child missions, etc.	Pass
	Mission Asset Assignment	Unit	Test	Create a unit test that executes the assignment of the mission to the assets and ensure that the known most appropriate asset is allocated to the mission. Vary the asset pool with different priorities, sensor types, and distances from objective.	Pass
	Cover AOI Mission Process	Integration	Test/ Demonstration	Create a demonstration test for the cover AOI mission to check flow of actions and completion.	Pass
	ID Track Mission Process	Integration	Test/ Demonstration	Create a unit test for the ID Track mission to check flow of actions and completion.	Pass
	Move Mission Process	Integration	Test/ Demonstration	Create a unit test for the move mission to check flow of actions and completion.	Pass
	MTI Mission Process	Integration	Test/ Demonstration	Create a unit test for the MTI coverage mission to check flow of actions and completion.	Pass
	SIGINT Mission Process	Integration	Test/ Demonstration	Create a unit test for the SIGINT coverage mission to check flow of actions and completion.	Pass

Category	Test Item	Type	Method	Description	Result
Missions and Behaviors	Track Target Mission Process	Integration	Test/ Demonstration	Create a unit test for the track target mission to check flow of actions and completion.	Pass
	SAR Mission Process	Integration	Test/ Demonstration	Create a unit test for each mission to check flow of actions and completion.	Pass
	Find Track Mission Process	Integration	Test/ Demonstration	Create a unit test for each mission to check flow of actions and completion.	Pass
	HUMINT Mission Process	Integration	Test/ Demonstration	Create a unit test for each mission to check flow of actions and completion.	Pass
	Convoy Support Mission Process	Integration	Test/ Demonstration	Create a unit test for each mission to check flow of actions and completion.	Pass
	New Track Detected Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	Time Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	Track Filter Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	Random Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	HUMINT Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	Message Link Condition Check	Unit	Test	Create a unit test to set up the state of the simulation to run against the conditions. Vary the state to execute each possible path.	Pass
	Start Movement Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass
	Stop Movement Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass
	Scan Time Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass

Category	Test Item	Type	Method	Description	Result
Missions and Behaviors	Activate IED Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass
	Coercion Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass
	Create Track Event	Integration	Test/ Demonstration	Create unit tests or vignettes to verify the change in state of the simulation based on the execution of the event. Vary the parameters of the event if applicable. A vignette must be used if the event has a time based effect.	Pass
	LD Circle Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	LD Hover Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Circle around Perimeter Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Fixed Sensor Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Follow Waypoint List Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Hover in Circle Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Hover & Keep LOS Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Hover/Park Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Keep LOS Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Move Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Move to Track Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass

Category	Test Item	Type	Method	Description	Result
Missions and Behaviors	Moving Circle Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Perpendicular Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Remain at Location Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Remain at Standoff Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
	Serpentine Behavior Creation	Integration	Test/ Demonstration	Create a vignette to execute the behavior, compare the resulting waypoints and updating process with the expected results.	Pass
Analysis Toolkit	Record Processing	Unit	Test	Create a unit test to read a set of known records and match them to a set of known raw data types. Compare the executed results to the expected results.	Pass
	Raw Data Trace Creation	Unit	Test	Given an output file, create a trace for each raw data type. Compare the resulting trace to the expected result.	Pass
	Data Filter Checks	Unit	Test	Create a set of records and filters do conduct comparisons. Compare the expected result to the executed results of each filter checked. Run for all expected filter and record pairings.	Pass
	Data Element Creation	Integration	Test/ Demonstration	Create a data trace to be created with a set of known filters. Compare the trace levels to the expected result. Vary the number of levels and filters used.	Pass
	Data Operations	Unit	Test	For each operator, execute the operation on a set of known inputs and compare to the expected outcome.	Pass
	Data Export	Integration	Demonstration	Create an analysis session with known data values, export the data using both methods and compare the results to the expected output.	Pass
	Data Operator Element Creation	Integration	Test/ Demonstration	Create a set of data elements to perform operations on to exercise the use of single trace, trace lists, and binary data element operators.	Pass
	Coverage Mapping	Integration	Demonstration	Create several output files containing coverage data with known coverage locations. Use the toolkit to extract the data and use the interface to map the data on the terrain. Compare to the expected result.	Pass
	Density Mapping	Integration	Demonstration	Create an output file with known locations of interest. Plot the vector data using the interface and compare to the expected result.	Pass

Category	Test Item	Type	Method	Description	Result
Other Simulation Types	Coverage Analysis Simulation	Integration	Demonstration	Create a series of coverage simulations to examine the changes in the output based on the different coverage behaviors. Vary the behaviors from simple (only a few areas looked at) to complex (several areas looked at several times). Compare the resulting coverage images to what the expected coverage is.	Pass
	Terrain Analysis Simulation	Integration	Demonstration	Create a series of terrain simulation to examine the changes in the output based on the terrain features and sensor locations. Vary the features from simple (a single building) to complex (many buildings/mountains) and vary the number of sensor locations. Compare the images/metrics to the expected patterns.	Pass

17. CONFIGURATION MANAGEMENT

Configuration Management (CM) is a systems engineering process for establishing and maintaining consistency of a product's performance, functional and physical attributes with its requirements, design and operational information throughout its life. The FOCUS CM process consists of an overall software development method, a source control application for managing the code and work process, and full set of documentation for FOCUS.

17.1 Scrum Agile Software Development Method Overview. The FOCUS software development process is a modified version of the agile software development method known as “Scrum.” Due to the rapid development of new technologies and the constant change of system performance and procedures, an agile method is the appropriate development method. Agile methods generally promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages team work, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals. Scrum is a development method that is optimized for small teams and has the following principles:

- Iterative development with small increments
 - Frequent inspection and adaptation
 - Working product after each iteration
- Requirements, technology, and capabilities emerge over time
- Self-organization, adaptation in response to what emerges
 - Empowered, co-located, cross-functional, collaborating and accountable teams
 - Constant visibility into the project for all parties
- Continual process improvement via retrospectives

17.2 Roles. The Scrum process defines the following roles in the team:
Scrum Master (Team Leader)

- The person responsible for the correct implementation of the Scrum process and for maximizing its benefits
- Provides leadership, guidance, and coaching for the Scrum team
- Removes impediments to progress
- Removing the barriers between development and the customer so the customer directly drives development
- Improving the lives of the development team by facilitating creativity and empowerment
- Improving the productivity of the development team in any way possible

Product Owner (Lead Developer)

- Domain Expert
- Establishes, nurtures, and communicates product vision

- Manages (e.g., sets priorities for) project requirements and Return on Investment (ROI) in the form of a Product Backlog; disaggregates product features into Sprint-sized (or less) “stories”
- Performs iteration and release planning; computes metrics (e.g., velocity, conditions of satisfaction)

Development Team

- Cross-functional with no strictly-defined roles (Designers, developers, Quality Assurance (QA)/testers, etc.)
- Self-organizing to meet needs as they arise
- Self-managing: empowered to solve their own problems

17.3 Process Flow. The Scrum development process defines iterations known as “Sprints” which are a defined number of days. The FOCUS development team is currently using a monthly Sprint model with CM meetings on the first Tuesday of each month. As seen in Figure 9, the Scrum process is an iteration of requirements discovery, development of items into a product backlog, planning of Sprint tasks, execution of tasks during a Sprint, and reviewing the Sprint.

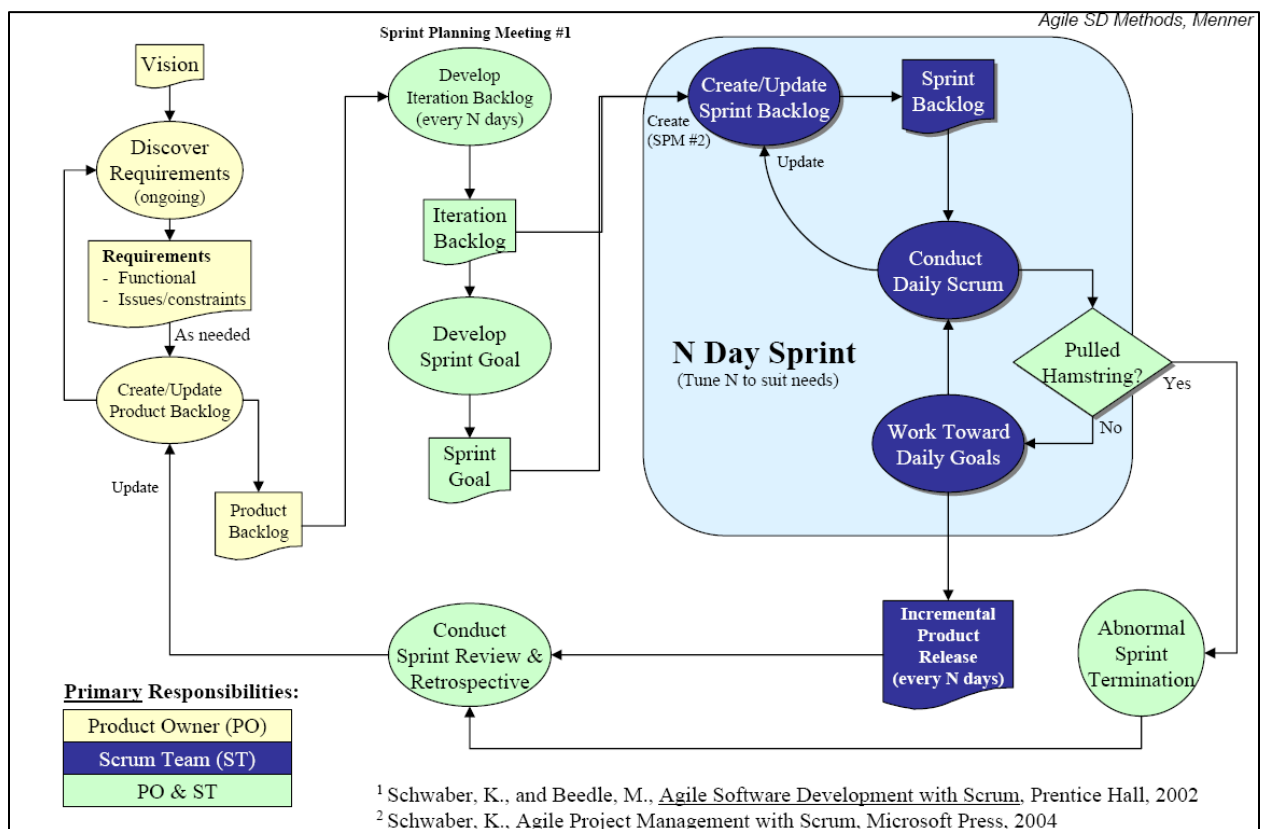


Figure 9: Software Development Process

17.4 Artifacts. The Scrum process generates the following artifacts to track progress:

Product Backlog

- A list of functional and non-functional requirements for the objective system
 - Functionality-based (“implement tracking”)
 - Story-based (“let user choose view”)
 - Task-based (“fix bug that ... ”)
 - Combinations of above
- Prioritized so top items are most likely to generate value
- Contains issues or constraints that must be satisfied before other items can be completed
- Level of Effort (LOE) estimates for each item

Sprint Backlog

- A list of tasks to be completed during the Sprint
 - Assigned to a single development team member
 - Each task has a LOE
 - Team members track estimated time remaining on task
- A product backlog item is typically broken into several smaller tasks to be completed during the Sprint

17.5 Implementation using Team Foundation Server. The Scrum artifacts are automated in the Visual Studio Team Foundation Server (TFS) Source Control System. TFS keeps a database of all current backlog, or “work,” items where the product backlog items are “User Stories” and sprint backlog items are tasks, issues, tests, or bugs. TFS is an integrated part of the Visual Studio Team System that automatically tracks the progress of the team through the creation and completion of work items. Figure 10 depicts a list of user story work items that make up the product backlog. This view shows the iteration the work item is associated with, the number of story points (i.e., LOE), and the state of the work item.

ID	Stack Rank ▲	Story Poi...	Title	State	Iteration Path	Work It...
27		2	Standoff Chemical Detector	Resolved	FOCUS	User Story
28		2	Point Chemical Detector	Resolved	FOCUS	User Story
30		2	Add Image Overlay GUI using images	Resolved	FOCUS\December11	User Story
32		3	P(detect) Simulation	Resolved	FOCUS\December11	User Story
33		1	Keep LOS Behavior (non-hovering)	Resolved	FOCUS\December11	User Story
34		1	Create Track Mission Event	Resolved	FOCUS\December11	User Story
35		1	Normal Extinction Coefficient using database	Resolved	FOCUS\December11	User Story
37		3	Sensor Missions - ID Track, Track Target	Resolved	FOCUS\January12	User Story
39		3	New observation condition	Resolved	FOCUS\January12	User Story
41			Library Form Inheritance	Active	FOCUS	User Story
44		2	TTPM Importer	Resolved	FOCUS\February12	User Story
45		1	Laser Designation Updates	Resolved	FOCUS\February12	User Story
46		3	Fusion Updates	Active	FOCUS\February12	User Story

Figure 10: Product Backlog in Team Foundation Server

During monthly meetings, the selected work items can be assigned to the Sprint and a team member. When the team member completes the task, the code is checked into the project and the work item is associated with the code change set. Team members have

the ability to create subtask work items as they see fit in order to complete a user story item.

A user story work item contain descriptions of the task, links to the change sets associated with the tasks, links to the test cases for the tasks, and can contain attachments of external files associated with the task. Figure 11 depicts the interface for updating a work item.

Figure 11: Work Item in Team Foundation Server

17.6 V&V as part of the Software Development Process. V&V is an essential part of the software development process that is integrated into all phases of development. The Scrum method allows for the integration of V&V processes with a “definition of done.” A work item is not complete until specific documentation and reviews have been completed. Figure 12 shows the development processes and the associated documentation and reviews. TFS also allows code to be reviewed by another team member before being “checked in” to the baseline version.

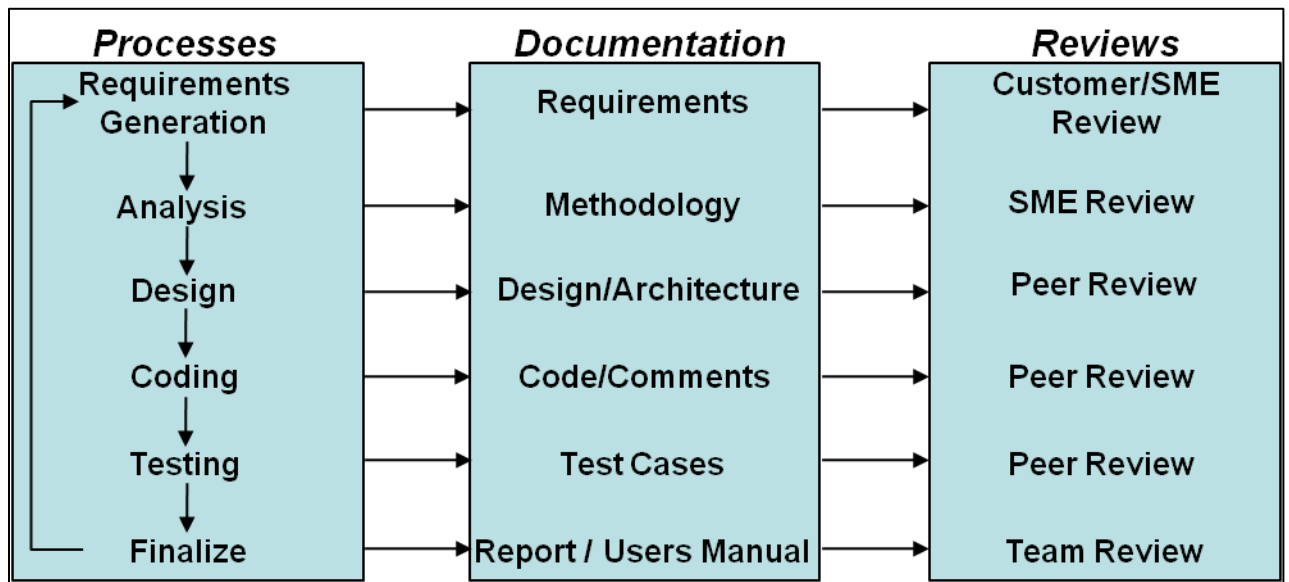


Figure 12: Processes - Documentation - Reviews

18. ARMY AND JOINT C4ISR STUDY SUPPORT

Section removed from public release version.

19. SUMMARY

19.1 V&V Synopsis. Results from the FOCUS Version 1.0 V&V demonstrate the stability, credibility, and accuracy of the algorithms, equations, and data utilized by the FOCUS Version 1.0. Any software issues that were identified during V&V have been corrected by the FOCUS development team at AMSAA. FOCUS V&V is an ongoing process and as new algorithms are added, V&V will be completed and documented. Currently, the fusion algorithms are being improved and tested in coordination with CMIF and will be documented when completed.

Table 3: V&V Results and Approach Summary

Model Areas for V&V	General Results	SME Validation	Stand-Alone Benchmark	Sensitivity Analysis	Demo Through Study	Unit Testing	Integration Testing
Simulation Architecture	Pass	n/a	n/a	n/a	✓	✓	✓
Environment	Pass	n/a	n/a	n/a	✓	✓	✓
Entity Movement	Pass	n/a	n/a	n/a	✓	✓	✓
Sensors							
<i>EO/IR</i>	Pass	✓	✓	✓	✓	✓	✓
<i>Laser Designation</i>	Pass	✓	✓	✓	✓	✓	✓
<i>MTI Radar</i>	Pass	✓	✓	✓	✓	✓	✓
<i>SAR</i>	Pass	✓	✓	✓	✓	✓	✓
<i>SIGINT</i>	Pass	✓	✓	✓	✓	✓	✓
<i>HUMINT</i>	Pass					✓	✓
Multi-Sensor Data Fusion	Future						
Communications/ RF Propagation	Pass	✓	✓		✓	✓	✓
Missions	Pass	n/a	n/a	n/a	✓	✓	✓
Analysis Toolkit	Pass	n/a	n/a	n/a	✓	✓	✓
Mini Simulations	Pass	n/a	n/a	n/a	✓	✓	✓

19.2 Application. FOCUS is intended to simulate the tactical performance of ISR systems and to permit rapid analysis and interpretation of simulation data. Simulation of events down to platform-level resolution (vehicle, aircraft, dismounted soldier, etc.) as well as behaviors such as movement, collection, acquisition, and communications in robust code modules will be used to predict overall performance of ISR systems of systems. FOCUS has been used to support various Army and Joint

studies. The V&V documentation is intended to support the accreditation of FOCUS for the following purposes:

- ✓ Perform comparative sensor performance analyses between ISR systems across the following domains:
 - Electro-Optical/Infrared
 - Moving Target Indicator Radar (Ground and Dismount)
 - Synthetic Aperture Radar
 - Signals Intelligence (Communication and Electronic)
- Sensitive to the following operational conditions:
 - Terrain
 - Environmental conditions
 - Threat make-up and behaviors
 - ISR system TTP (movement, search pattern, cueing, etc.).
- ✓ Perform comparative sensor coverage analysis between EO/IR system(s) given a set of TTPs.
- ✓ Perform line of sight and/or sensor performance analysis over a given terrain and sensor location(s).
- ✓ Perform comparative analysis between laser designation systems on the ability to keep the laser spot on a target.

19.3 Conclusions. Through an extensive series of SME reviews, detailed software inspections, unit and integration tests, and parametric analyses, FOCUS Version 1.0 has been shown to perform as expected and produce valid results within its analytical realm. The development team has implemented a process for the continued evolution of the simulation for use in future Joint and Army studies.

REFERENCES

1. *Management of Army Models and Simulations*. Department of the Army. AR 5-11.
2. *Verification, Validation and Accreditation of Army Modeling and Simulation*. Department of the Army. DA PAM 5-11. 1999.

THIS PAGE INTENTIONALLY LEFT BLANK.